

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA CIVIL Y AMBIENTAL

**EXPLORACIÓN DEL PRODUCTO SATELITAL DE PRECIPITACIÓN
IMERG V06 COMO INSUMO PARA LA HIDROLOGÍA**

**DETERMINACIÓN DE CURVAS INTENSIDAD-DURACIÓN-
FRECUENCIA (IDF) SUB-DIARIA Y DIARIA**

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO
REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
INGENIERÍA CIVIL**

LUIS MARCELO VARGAS DOMÍNGUEZ

luis.vargas02@epn.edu.ec

DIRECTOR: PhD. CARLOS SEBASTIÁN PÁEZ BIMOS

carlos.paezb@epn.edu.ec

DMQ, agosto 2023

CERTIFICACIONES

Yo, LUIS MARCELO VARGAS DOMÍNGUEZ declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

LUIS MARCELO VARGAS DOMÍNGUEZ

Certifico que el presente trabajo de integración curricular fue desarrollado por LUIS MARCELO VARGAS DOMÍNGUEZ, bajo mi supervisión.

CARLOS SEBASTIÁN PÁEZ BIMOS PhD.

DIRECTOR

DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

LUIS MARCELO VARGAS DOMÍNGUEZ

PhD. CARLOS SEBASTIÁN PÁEZ BIMOS

DEDICATORIA

A Dios por darme la vida y manejar los hilos a favor de tener la oportunidad de superarme.

A mis padres Inés y Marcelo, por su gran esfuerzo al solventar este sueño, valor al creer en mis capacidades y determinación al nunca dejarse vencer por las dificultades. Todo esto sirvió de inspiración para vencer todas las dificultades que se me presentaron en este magnífico camino.

A mi hermana Verito por ser una guía y brújula en mi vida.

A mi hermanita Alexita, mi reflejo por ser un gran apoyo.

A mi familia por tener siempre una palabra reconfortante.

A mis abuelos Rosa y Carlos por darme ejemplo de fuerza, integridad y sabiduría en componer cualquier inconveniente.

A mis angelitos, mami Fabi y papá Abelito su recuerdo, enseñanzas y los momentos maravillosos que pasamos los llevare siempre en mi corazón.

A mis amigos por haber compartido lindos momentos.

A mis héroes, personas que me incentivan, y demás personas que se sienten parte de este proceso.

AGRADECIMIENTO

A mí, por intentarlo, por dar siempre lo mejor, por la perseverancia, por la entrega y por tener la resiliencia necesaria.

A mis padres por su amor incondicional.

Agradezco a la Empresa Pública Metropolitana de Agua Potable y Saneamiento de Quito (EPMAPS) y al Instituto Nacional de Hidrología y Meteorología (INAMHI), por compartirme la información para el desarrollo de este trabajo.

A mi tutor el Ing. Sebastián Páez, PhD, por su gran paciencia, por compartir sus amplios conocimientos y por su incomparable ayuda en este proceso, que sin su apoyo nada de esto hubiera sido posible.

ÍNDICE DE CONTENIDO

CERTIFICACIONES.....	I
DECLARACIÓN DE AUTORÍA	II
DEDICATORIA.....	III
AGRADECIMIENTO.....	IV
ÍNDICE DE CONTENIDO.....	V
ÍNDICE DE FIGURAS	VII
ÍNDICE DE TABLAS.....	VIII
ÍNDICE DE ANEXOS	X
1 DESCRIPCIÓN DEL COMPONENTE DESARROLLADO	1
1.1 Objetivo general.....	2
1.2 Objetivos específicos.....	2
1.3 Alcance	3
1.4 Marco teórico	3
1.4.1 Productos Satelitales.....	4
1.4.2 GPM.....	5
1.4.3 IMERG	5
1.4.4 Validación de datos	7
1.4.5 Series máximas anuales (AMS).....	10
1.4.6 Ajuste de distribución de frecuencias	10
1.4.7 Prueba de bondad de ajuste	17
1.4.8 Periodo de retorno.....	19
1.4.9 Curvas IDF.....	19
2 METODOLOGÍA.....	20
2.1 Metodología	22
2.1.1 Recopilación de información	24

2.1.2	Lectura de datos satelitales.....	27
2.1.3	Series de datos.....	28
2.1.4	Validación IMERG	30
2.1.5	Series máximas anuales	32
2.1.6	Ajuste de frecuencia	32
2.1.7	Test de bondad de ajuste.....	34
2.1.8	Periodo de retorno.....	34
2.1.9	Curvas IDF.....	34
3	RESULTADOS, CONCLUSIONES Y RECOMENDACIONES.....	35
3.1	Resultados	35
3.1.1	Series temporales.....	35
3.1.2	Validación IMERG V06.....	38
3.1.3	Series máximas anuales	40
3.1.4	Análisis del ajuste funciones de distribución de probabilidad	46
3.1.5	Curvas IDF.....	48
3.2	Discusión.....	65
3.3	Conclusiones.....	66
3.4	Recomendaciones	68
4	REFERENCIAS BIBLIOGRÁFICAS	68
5	ANEXOS.....	73

ÍNDICE DE FIGURAS

Figura 1. Área cubierta por GPM (celeste) y TRMM (beige)	4
Figura 2. Red Internacional de Satélites de la misión GPM.	5
Figura 3. Mapa geográfico de las estaciones meteorológicas utilizadas en el Ecuador. .	20
Figura 4. Diagrama de flujo de la metodología empleada	23
Figura 5. Serie de Precipitación Máxima Anual (2000-2021)	29
Figura 6. Series temporales de precipitación mensuales Observadas vs IMERG V06. ...	36
Figura 7. Validación con Coeficiente de correlación de Pearson, RMSE, PBIAS y NSE de los datos IMERG V06 vs Observados para distintas duraciones.	38
Figura 8. Serie Máxima Anual con datos de 30 minutos IMERG V06 vs Observadas.	41
Figura 9. Serie Máxima Anual con datos de 1 hora IMERG V06 vs Observadas.	41
Figura 10. Serie Máxima Anual con datos de 2 horas IMERG V06 vs Observadas.	42
Figura 11. Serie Máxima Anual con datos de 6 horas IMERG V06 vs Observadas.	42
Figura 12. Serie Máxima Anual con datos de 12 horas IMERG V06 vs Observadas.	43
Figura 13. Serie Máxima Anual con datos de 24 horas IMERG V06 vs Observadas.	43
Figura 14. Curvas IDF para la estación M0003 IMERG V06 vs Observada.	49
Figura 15. Curvas IDF para la estación M0006 IMERG V06 vs Observada.	50
Figura 16. Curvas IDF para la estación M0007 IMERG V06 vs Observada.	51
Figura 17. Gráfico N6. Curvas IDF para la estación M0025 IMERG V06 vs Observada.	52
Figura 18. Curvas IDF para la estación M033 IMERG V06 vs Observada.	53
Figura 19. Curvas IDF para la estación C15 IMERG V06 vs Observada.	54
Figura 20. Curvas IDF para la estación P12-P27 IMERG V06 vs Observada.	55
Figura 21. Gráfico N6. Curvas IDF para la estación C05-P09-P70 IMERG V06 vs Observada.	56
Figura 22. Gráfico N6. Curvas IDF para las estaciones C02-C04-C14-P03-P08-P11-P28-P31-P32 IMERG V06 vs Observada.	58

ÍNDICE DE TABLAS

Tabla 1. Características de los datos de precipitación IMERG.....	7
Tabla 2. Valores de referencia de PBIAS y NSE y su relación con el desempeño para modelos hidrológicos.....	9
Tabla 3. CDF y PDF de Log Normal.....	11
Tabla 4. CDF y PDF de Log Normal de tres parámetros.....	12
Tabla 5. CDF y PDF de Gumbel.....	12
Tabla 6. CDF y PDF de Pearson tipo III.....	13
Tabla 7. CDF y PDF de Log Pearson tipo III.....	14
Tabla 8. Ecuaciones de método de momentos ordinarios.....	15
Tabla 9. Regiones del Ecuador con su dependencia regidora de precipitación.....	21
Tabla 10. Lista de las variables de los datos de salida IMERG V06 de media hora.....	24
Tabla 11. Periodo de registro de datos de las estaciones del INAMHI.....	26
Tabla 12. Periodo de registro de datos de las estaciones del EPMAPS.....	26
Tabla 13. Características de las estaciones utilizadas.....	27
Tabla 14. Función de distribución que mejor se ajusta por la prueba Kolmogórov-Smirnov.	46
Tabla 15. Intensidades en mm/h para periodos de retorno (Frecuencias) para cada duración, datos de curvas IDF para la estación M0003 IMERG V06 vs Observada.....	49
Tabla 16. Intensidades en mm/h para periodos de retorno (Frecuencias) para cada duración, datos de curvas IDF para la estación M0006 IMERG V06 vs Observada.....	50
Tabla 17. Intensidades en mm/h para periodos de retorno (Frecuencias) para cada duración, datos de curvas IDF para la estación M0007 IMERG V06 vs Observada.....	51
Tabla 18. Intensidades en mm/h para periodos de retorno (Frecuencias) para cada duración, datos de curvas IDF para la estación M0025 IMERG V06 vs Observada.....	51
Tabla 19. Intensidades en mm/h para periodos de retorno (Frecuencias) para cada duración, datos de curvas IDF para la estación M0033 IMERG V06 vs Observada.....	52
Tabla 20. Intensidades en mm/h para periodos de retorno (Frecuencias) para cada duración, datos de curvas IDF para la estación C15 IMERG V06 vs Observada.....	53
Tabla 21. Intensidades en mm/h para periodos de retorno (Frecuencias) para cada duración, datos de curvas IDF para la estación P12-P27 IMERG V06 vs Observada.....	54
Tabla 22. Intensidades en mm/h para periodos de retorno (Frecuencias) para cada duración, datos de curvas IDF para la estación C05-P09-P70 IMERG V06 vs Observada.	55

Tabla 23. Intensidades en mm/h para periodos de retorno (Frecuencias) para cada duración, datos de curvas IDF para la estación C02-C04-C14-P03-P08-P11-P28-P31-P32 IMERG V06 vs Observada.	56
Tabla 24. Error porcentual de las curvas IDF IMERG V06 respecto a la curvas observadas para cada una de las estaciones.	58

ÍNDICE DE ANEXOS

Anexo 1. Manual de descarga del producto IMERG V06.....	73
Anexo 2. Script Abrir IMERG V06	94
Anexo 3. Script agregación para las series temporales a diferentes duraciones	101
Anexo 4. Script de gráficas de series temporales.....	127
Anexo 5. Script de la validación de los datos	149
Anexo 6. Análisis CDF para la serie de 1hr	160
Anexo 7. Script gráfico de curvas IDF	170

RESUMEN

La escasez de datos pluviométricos en países en vías de desarrollo hace que los productos de precipitación basados en satélites jueguen un papel muy importante para el análisis hidrológico y el diseño de obras hidráulicas. En este estudio se presenta una evaluación del producto final de precipitación IMERG V06 con resolución espacial de $0.1^{\circ} \times 0.1^{\circ}$ en el periodo de (2000-2001) para la construcción de curvas Intensidad-Duración-Frecuencia (IDF). El área de estudio se centra en el Distrito Metropolitano de Quito (EPMAPS) y estaciones localizadas en los cuatro puntos cardinales de Ecuador (INAMHI). En base a las series de precipitación anual máxima en diferentes duraciones se realizó un ajuste a las distribuciones más comúnmente utilizadas: Gumbel, Log Normal, Pearson III y Log Pearson III, seleccionando el mejor ajuste por medio de pruebas de bondad: Chi-cuadrado y Kolmogórov-Smirnov. A partir de la función de distribución con mejor ajuste se obtuvieron las curvas IDF. En general, se determinó que el producto IMERG V06 subestima las intensidades para duraciones cortas y sobreestima las intensidades para duraciones largas. En la región Costa, las curvas IDF satelitales resultaron más similares a las observadas que en las demás regiones. En la región Sierra la comparación fue minuciosa debido a que en esta región se presenta más de una estación en un píxel, siendo las curvas más representativas las que se encuentran a mayor altitud; mientras que en la Amazonia se obtuvo resultados pocos satisfactorios.

PALABRAS CLAVE: curvas IDF; IMERG V06; serie máxima anual; Ecuador; función de distribución de probabilidad.

ABSTRACT

The scarcity of rainfall data in developing countries makes satellite-based precipitation products play a very important role for hydrological analysis and hydraulic design. In this study, an evaluation of the final IMERG V06 precipitation product with $0.1^\circ \times 0.1^\circ$ spatial resolution over the period of (2000-2001) is presented for the construction of Intensity-Duration-Frequency (IDF) curves. The study area is located in the Metropolitan District of Quito (EPMAPS) and stations located in the four cardinal points of Ecuador (INAMHI). Based on annual maximum precipitation series at different durations, fits were made to the most used distributions: Gumbel, Log Normal, Pearson III and Log Pearson III, selecting the best fit by means of goodness-of-fit tests: Chi-square and Kolmogorov-Smirnov. IDF curves were obtained from the distribution function with the best fit. In general, it was found that the IMERG V06 product underestimates intensities for short durations and overestimates intensities for long durations. In the Coast region, the satellite IDF curves were more similar to observations than those observed in the other regions. In the Sierra region, the comparison was detailed since here there is more than one station in a pixel, being the most representative curves those located at higher altitudes, finally in the Amazon region, the results were not very satisfactory.

KEYWORDS: IDF curves; IMERG V06; annual maximum series; Ecuador; probability distribution function.

1 DESCRIPCIÓN DEL COMPONENTE DESARROLLADO

La información hidrológica es usada para el desarrollo del diseño de estructuras hidráulicas como drenajes pluviales, azudes, presas, embalses y demás, buscan controlar inundaciones y aprovechar los recursos para generar energía. La escasez de datos pluviométricos en países en vías de desarrollo hace que los productos de precipitación basados en satélites jueguen un papel muy importante para el análisis hidrológico y posteriormente el diseño de obras hidráulicas. Por ejemplo: la tormenta de diseño obtenida de las curvas de intensidad-duración-frecuencia (IDF) es el insumo principal del diseño hidráulico para el control de inundaciones (Lau & Behrangi, 2022).

Las curvas de intensidad-duración-frecuencia (IDF) son representaciones gráficas que resumen las probabilidades condicionales (frecuencias) de duraciones promedio de lluvia e intensidades (Lau & Behrangi, 2022). Las curvas IDF se obtienen después de realizar mediciones pluviométricas, de esta información se selecciona las intensidades máximas históricas las cuales reciben un ajuste por medio de una función de probabilidad para un periodo de retorno. Sin embargo, en las mediciones existen ciertas limitaciones como: distribución irregular de la lluvia, efectos del viento, evaporación, tipos de precipitación, bloqueo de vista por obstáculos, errores de instrumentación, dificultades en áreas remotas, limitaciones temporales y calibración. (Salio et al., 2015).

Debido a estas limitaciones se ha analizado el uso de Productos de Precipitación Satelital (PPS), como: Integrated Multi-satellite Retrievals (IMERG) obtenido de la misión de precipitación mundial (GPM). GPM es una constelación de satélites integrados que está diseñado para compensar el muestreo restringido de lluvia en campo. El producto IMERG parte de la información disponible en satélites, utilizando estimaciones infrarrojas (IR) geo sincrónica de la órbita terrestre (Huffman et al., 2017). Cuenta con una resolución temporal de 30 minutos, una resolución espacial $0.1^{\circ} \times 0.1^{\circ}$ (aproximadamente $10 \times 10 \text{ km}$), cobertura espacial global de 60°S a 60°N . IMERG está disponible en tres tipos de datos: temprano, tarde y final. IMERG supera a otros productos, porque subestima levemente la precipitación (Huffman et al., 2017).

Recientemente se ha investigado la posibilidad de usar el producto IMERG para la elaboración de curvas IDF a nivel mundial con Kyaw et al., 2022; Lau & Behrangi, 2022; Rodríguez-Sandoval et al., 2023. Lau & Behrangi, 2022 realizaron su estudio en regiones semiáridas Arizona-Estados Unidos y Feldbach-Austria. El estudio reveló que al utilizar IMERG, la serie máxima anual se ajusta mejor con AB-AMS (calculando el promedio de

precipitación después de la serie máxima anual) que con AA-AMS (calculando el promedio de precipitación antes de la serie máxima anual). Por último, se encontró que el uso de la distribución de Gumbel en el cálculo de las curvas IDF de IMERG da como resultado una mejor concordancia en la prueba de bondad que las otras distribuciones de probabilidad. En Kyaw et al., 2022 realizaron su estudio en Yangon (Birmania), localizada en el sudeste asiático. Los resultados mostraron que la distribución generalizada de valores extremos (GEV) se ajusta mejor a la distribución de lluvia por hora de los datos satelitales. El resultado obtenido fue que IMERG es el producto más adecuado para construir las curvas IDF de Yangon entre GSMaP_NRT, GSMaP_GC, IMERG y TRMM. En este caso la distribución de Valor Extremo Generalizada (GEV) era la que mejor se ajustaba a la distribución horaria de las precipitaciones de los datos por satélite, además las curvas IDF con corrección de sesgo indicaron una mayor intensidad en las estaciones costeras que en las del interior. Rodríguez-Sandoval et al., 2023 realizó su estudio en Colombia donde se encontró que el producto IMERG subestima las intensidades de lluvia para las duraciones cortas y sobreestimaba las intensidades para largas duraciones. Los resultados también sugieren que el producto IMERG en este momento no es capaz de capturar la distribución de lluvia diaria en la mayoría de las estaciones. Sin embargo, para cerca del 20% de los pluviómetros, el análisis mostró que las intensidades máximas derivadas del IMERG se encuentran dentro de un error relativo de +/-25%, en comparación con las calculadas mediante el enfoque tradicional. En este contexto, se propone hacer la evaluación del producto IMERG V06 para la construcción de curvas IDF en Ecuador.

1.1 Objetivo general

Evaluar las curvas de intensidad, duración y frecuencia (IDF) obtenidas de los datos de precipitación satelital IMERG V06, en base a observaciones de campo.

1.2 Objetivos específicos

1. Descargar, preparar y validar los datos de precipitación del producto IMERG V06.
2. Descargar, preparar y validar los datos de precipitación observados.
3. Obtener métricas estadísticas para evaluación de la precipitación de IMERG V06 en relación con las observaciones de campo.
4. Elaborar las curvas IDF mediante precipitación IMERG V06 y precipitación observada, mediante el ajuste de distribución de frecuencias.
5. Comparar las curvas IDF obtenidas de datos satelitales con las curvas IDF de los datos observados del INAMHI y EPMAPS.

1.3 Alcance

La precipitación observada se mide a través de estaciones pluviométricas, de las cuales podemos obtener las precipitaciones máximas anuales de ciertas duraciones para diferentes años. La institución encargada de gestionar las estaciones pluviométricas que están localizadas dentro de Ecuador es el Instituto Nacional de Meteorología e Hidrología (INAMHI). Las 5 estaciones por utilizar del INAMHI en este estudio están distribuidas en las regiones Sierra, Costa y Oriente a lo largo del país: Nuevo Rocafuerte (M0007), Izobamba (M0003), La Argelia (M0033), Pichilingue (M0006) y La Concordia (M0025). Adicionalmente, se analizará con las estaciones administradas por la Empresa Municipal Pública de Agua Potable y Saneamiento de Quito (EPMAPS). La precipitación satelital se obtuvo del producto IMERG V06 en su versión final, la mejor ya que toma en cuenta observaciones en campo.

A los datos tendremos que organizarlos y a través de un tratamiento obtendremos la precipitación máxima anual de cada año. Se realizará un análisis hidrológico en base a funciones de distribución de probabilidad con el fin de relacionar la magnitud y frecuencia de la precipitación. Calcularemos los parámetros de esta función, después utilizando una prueba de bondad determinaremos, si los datos se ajustan a la distribución de probabilidad elegida. Finalmente, trazaremos las gráficas y comparamos las curvas IDF del IMERG V06 con las curvas IDF del INAMHI y EPMAPS. Con esto, podremos evaluar la viabilidad del uso del producto IMERG V06 para una potencial aplicación para el diseño de proyectos hidráulicos.

1.4 Marco teórico

La falta de información hidrológica es un problema muy común en los países en vías de desarrollo, en particular en Ecuador. Las limitaciones en los datos se dan por algunas causas como: falta de estaciones hidrológicas y meteorológicas dificultando la recopilación de datos, nula inversión en tecnología, acceso restringido, fiabilidad, entre otras. La escasa información genera un gran inconveniente en la planificación, diseño, gestión y posterior operación de proyectos hidráulicos importantes para el desarrollo del país. (Vera et al., 2016). Otro problema importante es la alta variabilidad espacial y temporal de la precipitación en el Ecuador, que por lo general no es registrada por las redes de estaciones meteorológicas. Los datos hidrológicos satelitales pueden suplir estas necesidades volviéndose transcendentales, por esta razón es importante evaluar su desempeño en nuestro medio (Ovando et al., 2018).

1.4.1 Productos Satelitales

En las últimas décadas, ha existido un avance tecnológico sin precedente, el cual permitió el desarrollo de varios productos de precipitación satelital. Estos productos funcionan de dos maneras, la primera con observaciones de ondas infrarrojas calibradas y la segunda es una fusión de ondas microondas más infrarrojas (Chen & Li, 2016). Los sensores infrarrojos en satélites geoestacionarios entregan estimaciones de precipitación a altas resoluciones temporales (de hasta 15 minutos), en cambio los sensores mixtos (microondas e infrarrojos) son más utilizados para medir la precipitación debido a que las fuentes radiactivas son muy afines con las partículas de precipitación (Hou et al., 2014). Existen varios productos de precipitación satelital, como:

- CHIRPS (Climate Hazards Group)
- CMORPH (Climate Prediction Center Morphing technique)
- PERSIANN (Precipitation Estimation from Remotely Sensed Information using Artificial Neural Networks)
- TMPA (Tropical Rainfall Measuring Mission Multi-Satellite Precipitation Analysis)
- IMERG (Integrated Multi-satellite Retrievals for GPM) (Méndez Rivas, 2018).

El producto TMPA se desarrolló de la misión TRMM (Tropical Rainfall Measuring Mission) mediante una combinación de observaciones infrarrojas con microondas pasivas. TRMM fue la pionera en implementar observaciones de las precipitaciones tropicales y subtropicales, pero está en decadencia debido a 3 importantes defectos. El primero es imperfecciones al apreciar la precipitación sólida y la lluvia de baja intensidad (menor a 0,5 mm/h); la segunda, es que la regularidad en la toma de muestras es de 15 horas a 4 días, y finalmente la carencia de cobertura de latitudes altas (más allá de 35° N/S). La misión sucesora de TRMM, es Global Precipitation Measurement (GPM), cuenta con una mayor cobertura con 16 orbitas por día, cubriendo las latitudes medias y altas (65°S-65°N de latitud), lo que equivale al 80% del planeta cada tres horas (Huffman et al., 2017).

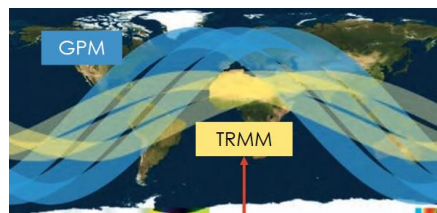


Figura 1. Área cubierta por GPM (celeste) y TRMM (beige)

Fuente: (Mehta & McCartney, 2020)

1.4.2 GPM

La misión Global Precipitation Measurement (GPM) se creó para resolver los inconvenientes con TRMM. El sistema GPM consigue estimar la precipitación sobre los trópicos y altitudes altas (65° N/S), incluyendo la precipitación sólida y poco intensa con una resolución temporal de 3 horas. (Boluwade et al., 2017). GPM consta de una red internacional de satélites que proporcionan la próxima generación observaciones globales de lluvia y nieve. La constelación GPM actual incluye 5 captadores de imágenes pasivos de órbita polar y 5 sondas de microondas pasivas de órbita polar. Su centro radica en el despliegue de un satélite "GPM Core" que lleva un sistema avanzado de radar/radiómetro para medir la precipitación desde el espacio y servir como estándar de referencia para unificar las mediciones de precipitación de una constelación de investigaciones y satélites operativos. GPM fue lanzado por la NASA (American National Aeronautics and Space) y la JAXA (Japanese Aerospace Exploration Agency) el 27 de febrero de 2014 y consta de 10 satélites (Huffman et al., 2017).



Figura 2. Red Internacional de Satélites de la misión GPM.

Fuente: (Mehta & McCartney, 2020)

1.4.3 IMERG

IMERG (Integrated Multi-satellite Retrievals for GPM) es un algoritmo que junta datos del GPM, GMI/DPR con los satélites de la constelación del GPM para producir excelentes estimaciones espaciales y temporales de precipitación. La principal meta del algoritmo IMERG es consolidar la continuidad con mayor precisión de la base de datos de precipitación iniciada con mediciones del TRMM (Huffman et al., 2019).

El algoritmo IMERG se basa en procesos de estimación de precipitación multisatélite desarrollados anteriormente: TMPA (Calibración Inter satelital con ajuste de la medición),

Técnica de Morphing del Centro de Predicción Climática (CPC), la NOAA con (CMORPH Interpolación del tiempo lagrangiana), PERSIANN de la Universidad de California, Irvine (calibración de microondas IR usando redes neuronales artificiales) y el sistema de procesamiento con ensamble de datos de entrada lluvia de la NASA (Méndez Rivas, 2018).

El IMERG necesita información de entrada que se obtiene de las mediciones de dos tipos de sensores diferentes: PMW (Modulación por ancho de pulso) e IR (Infrarrojo). El IMERG está esquematizado para arreglar las limitaciones de disponibilidad de datos PMW mediante el uso de la mayor cantidad de datos PMW y luego llena la información faltante con estimaciones IR de los satélites GEO (Grupo Especial de Operaciones), a través de una interpolación lineal basada en el movimiento característico que se obtiene de las imágenes IR de las plataformas GEO. Si los datos PMW son demasiado escasos, se utiliza el filtro de Kalman (Huffman et al., 2019).

IMERG ha presentado varias adaptaciones como la versión 03, la versión 05, la versión que se encuentra en fase de prueba la 07; mientras que la versión actualizada, estable y corregida de IMERG es la 06. En este estudio se utiliza la última versión de IMERG-final producto (V06) (Huffman et al., 2017). IMERG presenta tres presentaciones:

- Producto multisatélite “temprano” -4 horas- (actualmente 5) después del tiempo de observación.
- Producto multisatélite “tardío” -12 horas- (actualmente 16) después del tiempo de observación, y una vez recibido el análisis de calibre mensual.
- Producto de medición satelital “final” -2 meses- (actualmente 3,5) después del mes de observación.

La razón de estas tres presentaciones es de que después tomar la medida satelital se necesita hacer correcciones, las cuales deben satisfacer la demanda competitiva de puntualidad e integridad de los datos. La versión temprana da resultados relativamente rápidos para el análisis de inundaciones y otras aplicaciones de uso corto empleando solo Morphing hacia adelante. La versión tardía usa Morphing hacia adelante y hacia atrás con datos posteriores, y es apropiado usarlo para datos diarios como en la previsión de cosechas. La versión final presenta un análisis mensual, proporcionando resultados más precisos en regiones con falta de información, el uso recurrente de la versión final es la investigación (Huffman et al., 2019). En este estudio se usó la estimación de la presentación final, ocupando precipitaciones estimadas por múltiples sensores satelitales de la constelación GPM y ocupa los datos mensuales del GPCC para su calibración.

Cuenta con una latencia de 4 meses, y el periodo en el cual existen existe registro de datos es desde el 1 de junio del 2000 al 30 de septiembre del 2021.

Tabla 1. Características de los datos de precipitación IMERG.

Acrónimo	Nombre	Resolución Espacial	Resolución Temporal	Cobertura Global	Formato De Datos Salida
IMERG	Integrated Multisatélite Retrievals for GPM	0.1° x 0.1° aproximadamente 10 x 10 km	30 minutos	60°S a 60°N	HDF4, NetCDF, OpenDAP, ASCII, .gif, .png, KML.

Fuente: (Huffman et al., 2017).

1.4.4 Validación de datos

En algunos lapsos de tiempo la lluvia se detecta falsamente, o se omite. Para evaluar esta limitación se debe calcular algunas métricas estadísticas, que analizan los errores y diferencias con la precipitación observada en campo.

Coeficiente de Correlación de Pearson (CC)

Es utilizado para medir hasta qué punto dos variables cuantitativas están relacionadas linealmente, y que tan intensa es la correlación (Lalinde et al., 2018).

$$CC = \frac{\sum_{i=1}^n (IMERG_i - \overline{IMERG}) (OBSERVADA_i - \overline{OBSERVADA})}{\sqrt{\sum_{i=1}^n (IMERG_i - \overline{IMERG})^2} \sqrt{\sum_{i=1}^n (OBSERVADA_i - \overline{OBSERVADA})^2}} \quad \text{Ec. 1}$$

Donde:

CC = Coeficiente de correlación de Pearson.

$IMERG_i$ = Precipitación satelital de IMERG.

$OBSERVADA_i$ = Precipitación observada.

\overline{IMERG} = Promedios aritméticos de la precipitación satelital de IMERG.

$\overline{OBSERVADA}$ = Promedios aritméticos de la precipitación observada.

n = Número total de pasos del tiempo.

El rango del coeficiente Pearson es [-1 ; 1]. Entre más próximo es el valor del coeficiente de correlación a la unidad, mayor será la intensidad de la correlación entre variables. Si el signo de la correlación es positivo representa que la relación es directa, en cambio, si el signo es negativo la relación es inversa o indirecta (Loyola Cueva & Salazar Hinojosa, 2022).

Raíz del error cuadrático medio (RMSE)

RMSE mide la magnitud típica del error realizando una diferencia entre los valores estimados y los reales, elevándolos al cuadrado y calculando su promedio. Tiene la sensibilidad a errores más grandes y valores atípicos (Loyola Cueva & Salazar Hinojosa, 2022). La ecuación que se utiliza para calcularlo es la siguiente:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (IMERG_i - OBSERVADA_i)^2} \quad \text{Ec. 2}$$

Donde:

$RMSE$ = Raíz del Error Cuadrático Medio

$IMERG_i$ = Precipitación satelital de IMERG.

$OBSERVADA_i$ = Precipitación observada.

n = tamaño de la muestra

El valor perfecto del RMSE es 0 (Negrón Baez, 2014).

Sesgo relativo (PBIAS)

El término PBIAS proviene del inglés (Percent Bias) cual permite conocer el sesgo porcentual de los datos simulados con relación a las observaciones y es usado para determinar la tendencia promedio de los datos simulados a ser más grande o pequeña que los datos observados (Golmohammadi et al., 2014). Para calcularlo se utiliza la siguiente ecuación:

$$PBIAS = \frac{\sum_{i=1}^n (IMERG_i - OBSERVADA_i)}{\sum_{i=1}^n OBSERVADA_i} \times 100\% \quad \text{Ec. 3}$$

Donde:

$PBIAS$ = Sesgo relativo.

$IMERG_i$ = Precipitación satelital de IMERG.

$OBSERVADA_i$ = Precipitación observada.

n = tamaño de la muestra

El valor perfecto de PBIAS es 0, los valores positivos de PBIAS revelan sobreestimación del modelo, los negativos indican que existe subestimación del PBIAS en el modelo (Loyola Cueva & Salazar Hinojosa, 2022).

Eficiencia de Nash-Sutcliffe (NSE)

NSE es un índice estadístico normalizado que calcula el tamaño relativo de la varianza residual "ruido" en comparación con la varianza de los datos medidos "información". NSE indica qué tan bueno es el ajuste del gráfico a la línea 1:1 de datos observados frente a simulados (Nash & Sutcliffe, 1970).

$$NSE = \frac{\sum_{i=1}^n (OBSERVADA_i - \overline{OBSERVADA})^2 - \sum_{i=1}^n (IMERG_i - OBSERVADA_i)^2}{\sum_{i=1}^n (OBSERVADA_i - \overline{OBSERVADA})^2} \quad \text{Ec. 4}$$

Donde:

NSE = Eficiencia de Nash-Sutcliffe

$IMERG_i$ = Precipitación satelital de IMERG.

$OBSERVADA_i$ = Precipitación observada.

$\overline{OBSERVADA}$ = Promedios aritméticos de la precipitación observada.

n = tamaño de la muestra

Los valores de NSE (adimensionales) pueden variar de $-\infty$ a 1. Cuanto más cerca de 1, mayor es el ajuste entre los datos simulados y observados. Los resultados por debajo de 0 indican que los valores medios observados son más representativos que los valores predichos por el modelo (Cardoso De Salis et al., 2019). Los valores de referencia de los niveles de rendimiento se muestran en la siguiente tabla:

Tabla 2. Valores de referencia de PBIAS y NSE y su relación con el desempeño para modelos hidrológicos.

PBIAS (%)	NSE	Desempeño
0 a 10	0.75 a 1	Muy Buena
10 a 15	0.65 a 0.75	Buena
15 a 25	0.50 a 0.65	Mala
>25	<0.50	Inadecuada

Fuente: (Cardoso De Salis et al., 2019)

1.4.5 Series máximas anuales (AMS)

Las series de datos extremos pueden ser analizadas de dos maneras: Series máximas anuales (AMS) o series de duración parcial (PDS), (Instituto de Meteorología. Cuba & Planos Gutiérrez, 2015). Las series máximas anuales, son los valores máximos de precipitación que han sucedido anualmente en el periodo de registro. El análisis probabilístico por dos razones. El AMS es utilizado en el análisis probabilístico por dos cosas. La primera es por conveniencia, debido a que la gran parte de los datos se procesan de manera que la serie anual está disponible. La segunda es que hay una base teórica para extrapolar los datos de series anuales.

1.4.6 Ajuste de distribución de frecuencias

En ingeniería los procesos se los puede categorizar debido a la seguridad de su ocurrencia. Si la ocurrencia de las variables en un proceso es innegable, es decir si las variables siguen una ley determinada es un proceso determinístico. Por otra parte, si se toma en cuenta la probabilidad de ocurrencia y la falta de certeza existente, entonces es un proceso de naturaleza probabilística. La mayoría de los procesos en hidrología forman parte de la categoría probabilística (Morán, 2003).

Lo que significa que al experimentar algún proceso no es posible conocer la incertidumbre, pero se puede conocer todos los posibles resultados en la ejecución del experimento. Estos resultados se convertirán en nuestro espacio muestral. El modelo probabilístico hace uso de este espacio muestral, y otorga a cada uno de estos resultados una probabilidad de ocurrencia, es decir, una distribución de probabilidad con el uso de la funciones de probabilidad (Varas C. & Bois, 1998).

Distribuciones de probabilidad

Una distribución de probabilidad es una función que representa la probabilidad de ocurrencia de una variable aleatoria. Se puede realizar el resumen de una gran cantidad de información probabilística de manera compacta y de sus parámetros asociados, gracias al ajuste de una distribución (Chow et al., 2000).

Funciones de densidad de probabilidad

Una función de densidad de probabilidad (PDF) decreta la probabilidad de que una variable aleatoria tome un cierto valor. Para una variable aleatoria continua, no podemos usar un PDF directamente, ya que la probabilidad de que x tome cualquier valor exacto es cero (Singh & Chow, 2017).

Funciones de distribución acumulativa

Una función de distribución acumulativa (CDF) decreta la probabilidad de que una variable aleatoria tome un valor menor o igual que x (Singh & Chow, 2017). Para el análisis de frecuencia de series temporales de precipitación se utilizan varias funciones de distribución, como las siguientes:

Función Log Normal

La distribución de probabilidad Log-normal aparece siempre y cuando los factores causales sean multiplicativos. Chow fue el que inicio con el planteamiento de una ley de probabilidad logarítmica en hidrología, desde allí esta distribución se ha utilizado regularmente para el análisis de frecuencias de datos hidrológicos. Sea $Y = \log(X)$, $0 < x < \infty$. Si Y sigue una distribución normal, X sigue una distribución log-normal (Singh & Chow, 2017). Las fórmulas de la distribución se expresan en la siguiente tabla:

Tabla 3. CDF y PDF de Log Normal

Log Normal	
p.d.f	
$f(x) = \frac{1}{x\sigma(2\pi)^{\frac{1}{2}}} \exp\left(-\frac{(\ln x - u)^2}{2\sigma^2}\right)$	Ec. 5
$x > 0 \qquad u_x = \exp\left(u + \frac{\sigma^2}{2}\right)$	
Parámetros	
$u \in R$	$\sigma > 0$
c.d.f	
$F(x) = \Phi\left(\frac{\ln x - u}{\sigma}\right)$	Ec. 6
$\sigma = \exp(2u + \sigma^2)(\exp(\sigma^2) - 1)$	
$z = \left(\frac{\ln x - u}{\sigma}\right)$	
Fuente:(Singh & Chow, 2017)	

La distribución logarítmica normal de tres parámetros se usa con frecuencia en el análisis de frecuencia de inundaciones extremas, volúmenes de flujo estacional, curvas de duración del flujo, intensidad-duración de la lluvia, retención de agua del suelo, radio de poro y presión capilar de poro (Stedinger, 1980). Las fórmulas de la distribución se expresan en la siguiente tabla:

Tabla 4. CDF y PDF de Log Normal de tres parámetros

Log Normal de 3 parámetros	
p.d.f	
$f(x) = \frac{1}{(x-a)\sigma(2\pi)^{\frac{1}{2}}} \exp\left(-\frac{(\ln(x-a)-u)^2}{2\sigma^2}\right)$	Ec. 7
$x - a > 0$	
Parámetros	
$a, u \in R$	$\sigma > 0$
c.d.f	
$F(x) = \Phi\left(\frac{\ln(x-a)-u}{\sigma}\right)$ $z = \left(\frac{\ln(x-a)-u}{\sigma}\right)$	Ec. 8
Fuente:(Singh & Chow, 2017)	

Para las dos anteriores distribuciones, se tiene que:

u = media de Y .

σ = desviación estándar de Y .

z = variable estandarizada con la distribución normal.

Función Gumbel

Conocida también como distribución de valore extremos tipo I, es una de las distribuciones más conocidas para el análisis de frecuencias de extremos en inundaciones, sequías, precipitaciones y datos meteorológicos. Tomando el parámetro de forma $b = 0$, la distribución GEV converge a distribución EV I o Gumbel. Las fórmulas de la distribución se expresan en la siguiente tabla:

Tabla 5. CDF y PDF de Gumbel

Gumbel EV I	
p.d.f	

$$f(x) = \frac{1}{a} \exp\left(-\frac{(x-c)}{a} - \exp\left(-\frac{(x-c)}{a}\right)\right) \quad \text{Ec. 9}$$

$$x - c \geq 0$$

Parámetros	
$c \in \mathbf{R}$	$a > 0$

$$F(x) = \exp\left(-\exp\left(-\frac{x-c}{a}\right)\right) \quad \text{Ec. 10}$$

Fuente:(Singh & Chow, 2017)

Para la anterior distribución, se tiene que:

a = parámetro de escala

c = parámetro de localización.

Además, la distribución EVI tiene un coeficiente de asimetría constante (aproximadamente 1,14) y el coeficiente de curtosis constante (aproximadamente 5,4) (Singh & Chow, 2017).

Función Pearson III

La distribución de Pearson tipo III (PIII) es popular en el análisis de frecuencias de inundaciones y precipitaciones extremas, esta distribución es similar a una distribución gamma de tres parámetros (Singh & Chow, 2017). Las fórmulas de la distribución se expresan en la siguiente tabla:

Tabla 6. CDF y PDF de Pearson tipo III

Pearson Tipo III	
p.d.f	
$f(x) = \frac{1}{a\Gamma(b)} \left(\frac{x-c}{a}\right)^{b-1} \exp\left(-\frac{(x-c)}{a}\right)$	Ec. 11
$x \in [c, +\infty)$	

Parámetros	
$c \in \mathbf{R}$	$a, b > 0$

c.d.f	
$F(x) = \frac{1}{\Gamma(b)} \int_0^{\frac{x-c}{a}} y^{b-1} \exp(-y) dy$	Ec. 12
$y = \frac{x-c}{a}$	

Fuente:(Singh & Chow, 2017)

Donde:

a = parámetro de escala.

b = parámetro de forma.

c = parámetro de localización.

Función Log Pearson III

La distribución log-Pearson tipo III (LPIII) es una de las distribuciones más frecuentemente de las distribuciones más utilizadas para el análisis de frecuencias de extremos hidrológicos (Singh & Chow, 2017). Es la más utilizada en Estados Unidos por recomendación del (U.S. Water Resources Council, 1981). Si $Y = \ln X$ sigue la distribución PIII, X sigue la distribución LPIII. Las fórmulas de la distribución se expresan en la siguiente tabla:

Tabla 7. CDF y PDF de Log Pearson tipo III

Log Pearson Tipo III	
p.d.f	
$f(x) = \frac{1}{ax\Gamma(b)} \left(\frac{\ln x - c}{a}\right)^{b-1} \exp\left(-\frac{\ln x - c}{a}\right)$	Ec. 13
$x \in [\exp(c), +\infty)$	
Parámetros	
$c \in \mathbf{R}$	$a, b > 0$
c.d.f	
$F(x) = \frac{1}{\Gamma(b)} \int_0^{\frac{\ln x - c}{a}} y^{b-1} \exp(-y) dy$	Ec. 14
$y = \frac{\ln x - c}{a}$	

Fuente:(Singh & Chow, 2017)

Donde:

a = parámetro de escala.

b = parámetro de forma.

c = parámetro de localización.

Parámetros de las funciones de distribución

Cada una de las anteriores funciones de distribución, tiene distintos parámetros con características propias y dependientes de la distribución. Existen varios métodos para el cálculo de estos parámetros, pero se debe tener en cuenta que es obligatorio que los

parámetros sean eficientes, suficientes, consistentes y no sesgados. (Acosta Castellanos & Sierra Aponte, 2013). Entre los métodos más conocido tenemos:

Método De Los Momentos Ordinarios

Es un método que genera estimaciones sesgadas para poblaciones pequeñas. El método de los momentos ordinarios otorga una apreciación adecuada de los parámetros de una función de probabilidad, en la cual los momentos alrededor del origen son iguales a los momentos de la muestra (Chow et al., 2000). Los estimadores obtenidos por este método son asintóticamente eficiente, y más en distribuciones asimétricas, las más utilizadas en hidrología. Por eso se lo usa para un primer acercamiento en estimar los parámetros. (Koutsoyiannis, 2004). Este método sigue la siguientes expresiones:

Tabla 8. Ecuaciones de método de momentos ordinarios

Método de Momentos		
Momentos Observados	Momentos Observados centrados en cero	
$m_1 = \frac{1}{n} \sum_{i=1}^n X_i^1$	$u_1 = E [X^1]$	Ec. 15
$m_2 = \frac{1}{n} \sum_{i=1}^n X_i^2$	$u_2 = E [X^2]$	Ec. 16
⋮	⋮	
$m_k = \frac{1}{n} \sum_{i=1}^n X_i^k$	$u_k = E [X^k]$	Ec. 17
Y resolvemos el sistema de ecuaciones:		
$m_r = u_r$	$r = 1, \dots, k$	Ec. 18

Fuente: (Zozaya, 2019)

Método De Los Momentos Ponderados

El método de los momentos ponderados es parecido al de momentos ordinarios, pero con la diferencia de que pretende caracterizar la función de distribución por momentos ponderados de probabilidad, entonces solo se debe calcular los momentos ponderados por la probabilidad a partir de los datos muestrales (Acosta Castellanos & Sierra Aponte, 2013). Lleva la delantera sobre el método de los momentos ordinarios, debido a que se puede caracterizar varias funciones de distribución y además, es una técnica más firme ante la presencia de puntos extraordinarios en la muestra. Al ser una combinación lineal entre observaciones, no es necesario elevar al cuadrado o al cubo como sucede con el método ordinario (Pulgarín Dávila, 2009). Este método sigue la siguiente expresión:

$$m_{ijk} = E[x^i F^j (1 - F)^k] = \int_0^1 x(F)^i F^j (1 - F)^k dF \quad \text{Ec. 19}$$

Donde: i, j, k son números enteros. Para los casos en que $i=1$ y además $j=0$ o $k=0$, M_{10j} y M_{10k} son lineales en x . Cuando no existe muchos datos, este método es más capaz que los métodos de máxima verosimilitud y de momentos convencionales (Hosking et al., 1985).

Método De Máxima Verosimilitud

El método de máxima verosimilitud se encarga de encontrar el valor del parámetro que maximizara la probabilidad conjunta de ocurrencia observada en la distribución probabilidad. La función se maximiza al obtener las n derivadas parciales e igualándolas a cero, donde n es igual a la cantidad de parámetros que tiene la función. Por esta razón, tiene mayor precisión desde el punto de vista estadístico pero puede haber imprecisiones para muestras de menor tamaño comparado con las demás métodos, adicionalmente es de mayor laboriosidad desde el punto de vista de análisis de datos (Chavoshi, 2009).

Sea (X_1, \dots, X_n) una muestra aleatoria con una función de distribución $f(x|\theta)$.

La función de máxima verosimilitud puede ser definida como:

$$L(\underline{x}|\underline{\theta}) = L(X_1, \dots, X_n|\theta_1, \dots, \theta_n) = \prod_{i=1}^n f(X_i|\theta_1, \theta_2, \dots, \theta_k) \quad \text{Ec. 20}$$

$\underline{\theta}$ = Estimador de máxima verosimilitud es el valor que maximiza la función $L(\underline{x}|\underline{\theta})$.

Si la función de verosimilitud es diferenciable en $(\underline{\theta})$, el estimador de máxima verosimilitud EMV $\underline{\theta}$ es el que soluciona:

$$\frac{\partial}{\partial \theta_i} L(\underline{x}|\underline{\theta}) = 0, \quad i = 1, \dots, k \quad \text{Ec. 21}$$

Sin embargo, hay que chequear que:

$$\left[\frac{\partial^2}{\partial \theta_i \partial \theta_j} L(\underline{x}|\underline{\theta}) \right] \Big|_{\underline{\theta}=\hat{\underline{\theta}}} < 0, \quad i, j = 1, \dots, k \quad \text{Ec. 22}$$

A veces es muy complicado resolver la p.d.f. y ahí es más factible trabajar con la función logverosimilitud que se expresa así:

$$\lambda(\underline{x}|\underline{\theta}) = \ln L(\underline{x}|\underline{\theta}) = \sum_{i=1}^n \ln f(X_i|\theta_1, \theta_2, \dots, \theta_k) \quad \text{Ec. 23}$$

Y el EMV $\hat{\underline{\theta}}$ es igual a:

$$\frac{\partial}{\partial \theta_i} L(\underline{x}|\underline{\theta}) = 0, \quad i = 1, \dots, k \quad \text{Ec. 24}$$

1.4.7 Prueba de bondad de ajuste

Una prueba de bondad de ajuste ayuda a definir si la muestra se acomoda a la distribución de probabilidad que se está usando; es útil para hipótesis simples o compuestas. El objetivo es establecer si es factible utilizar una distribución determinada en una serie de datos (Acosta Castellanos & Sierra Aponte, 2013). Es preferible utilizar las pruebas de bondad de ajuste con métodos cuantitativos, que permiten una resolución objetiva sobre el ajuste. Las pruebas sobre modelos de distribución tienen las siguientes fases:

1. Se calcula un estadígrafo (números resúmenes, que permiten sacar conclusiones acerca de la estructura de una muestra de datos) con los datos observados.
2. Se determina la probabilidad de obtener el estadígrafo calculado, en el hipotético modelo correcto. La referencia es una tabla probabilística que da los percentiles del modelo de distribución del estadígrafo.
3. Si la probabilidad de lograr el valor calculado del estadígrafo es bajo se concluye que el modelo supuesto no arroja una adecuada representación de la muestra.

Este proceso autoriza desechar un modelo por no ser el idóneo, pero no permite probar que el modelo probabilístico sea el correcto. (Varas C. & Bois, 1998). Entre las pruebas más conocidas tenemos:

Test Kolmogórov-Smirnov

La prueba no es paramétrica, hace uso de la función de distribución acumulada empírica y teórica de las variables y se basa en la diferencia máxima absoluta de estas. Permite calcular la relación que existe entre una distribución de una muestra de datos y una distribución teórica, la prueba de bondad de ajuste se aprueba si el valor estadístico de la prueba de la muestra es menor que el valor tabulado para un nivel de probabilidad requerido Si tenemos dos poblaciones de tamaño $N = m + n$ que tiene observaciones $x_1, x_2, x_3, \dots, x_n$ e $y_1, y_2, y_3, \dots, y_m$ teniendo en cuenta que x e y son mutuamente independientes, x viene de la población continua P_1 e y de P_2 . La hipótesis nula es que

ambas distribuciones son iguales (muestra de la misma población). Hay que calcular el estadígrafo J que es el valor máximo de la diferencia absoluta entre dos funciones de distribución calculada (Varas C. & Bois, 1998), de la siguiente forma:

$$F_m(a) = \frac{\text{número de } X \leq a}{m} \quad \text{Ec. 25}$$

$$G_n(a) = \frac{\text{número de } Y \leq a}{n} \quad \text{Ec. 26}$$

$$J = \frac{m \cdot n}{d} \max_{-\alpha \leq a \leq \alpha} x e^{-x^2} \{|F_m(a) - G_n(a)|\} \quad \text{Ec. 27}$$

La dócima (prueba de hipótesis estadística) desecha la hipótesis nula si J es mayor o igual que un valor crítico que depende de m, n y del nivel de significancia, si los tamaños de muestras son grandes se puede usar:

$$Q(\lambda) = \text{Prob}(J < \lambda) = \sum_{j=-\infty}^{j=+\infty} (-1)^j \exp(-2j^2 \lambda^2) \quad \text{Ec. 28}$$

$$\lambda > 0$$

Donde:

m = tamaño de la muestra de la variable X.

n = tamaño de la muestra de la variable Y.

d = mínimo común múltiplo de m y n.

La dócima es rechazar H_0 si $J \geq q^*$

Donde:

$$q^* = Q(q^*) = 1 - \alpha \quad \text{Ec. 29}$$

Test Chi-Cuadrado

Es la prueba más utilizada y solo se puede aplicar si los parámetros son determinados por el método de máxima verosimilitud. Compara en partes antes definidas de la variable aleatoria, los casos observados con los teóricos. La ecuación que representa esta prueba es la siguiente:

$$X^2 = \sum_{i=1}^k \frac{(\text{OBSEVADAS}_i - \text{IMERG}_i)^2}{\text{IMERG}_i} \quad \text{Ec. 30}$$

Donde:

$OBSEVADAS_i$ = Frecuencias absolutas observadas.

$IMERG_i$ = Frecuencias absolutas teóricas.

X^2 = por lo general tiene una distribución Chi-cuadrado con:

$$\text{Grados de libertad} = K - S - 1$$

Ec. 31

K = Número de intervalos definido

S = Número de parámetros estimados en el modelo.

Para que el ajuste de la distribución a la muestra sea bueno, el valor de chi-cuadrado debe ser menor, o lo mucho, igual al valor teórico que la distribución chi-cuadrado para un nivel de significación del 5%. Con las tablas de distribución de chi-cuadrado se puede conocer el valor teórico de chi en función de lo K y S , se recomienda usar pocas clases para que el valor teórico de casos observados en cada clase sea mínimo 5 (Varas C. & Bois, 1998).

1.4.8 Periodo de retorno

El periodo de retorno de cualquier evento extremo (lluvias torrenciales, temperaturas extremas, huracanes, etc.), es decir, es el tiempo que en promedio se cree que será igualado o excedido, en otras palabras, es la frecuencia con la que se presenta un evento (Méllice & Reason, 2007). Puede ser que para un umbral (frecuencia extrema) de intensidad media, exista más de un valor que lo supere en un mismo año, con la AMS solo uno de ellos, el mayor, formara parte de la serie de observación, desechando lo demás, inclusive pudiendo ser mayor que los máximos de otros años (Fitzgerald, 2007).

1.4.9 Curvas IDF

Las curvas Intensidad – Duración – Frecuencia (IDF) son las curvas que relacionan los puntos representativos de la intensidad de la precipitación en intervalos de diferente duración, y correspondientes todos ellos a una misma frecuencia o período de retorno (Témez, 1978). Las curvas IDF señalan la probabilidad de ocurrencia de intensidades de precipitación en períodos cortos para distintas duraciones. Por lo general se grafica un conjunto de curvas, cada una mostrando una frecuencia de ocurrencia o período de retorno expresado en años (Varela et al., 2004).

2 METODOLOGÍA

La zona de estudio comprende a la ubicación de las estaciones pluviométricas utilizadas en la (Figura 3), llegando a abarcar después del análisis todo el territorio ecuatoriano. Las estaciones pluviométricas están distribuidas en las tres principales regiones del Ecuador continental: Costa con las provincias de Esmeraldas y Los Ríos, Sierra con las provincias de Pichincha y Loja, y Oriente con la provincia de Sucumbíos. La mayor concentración de estaciones se encuentra en el Distrito Metropolitano de Quito (DMQ), aquí se podrá realizar un análisis minucioso de los resultados, debido a que es posible efectuar una comparación entre dos o más estaciones que estén localizadas dentro del mismo píxel IMERG.

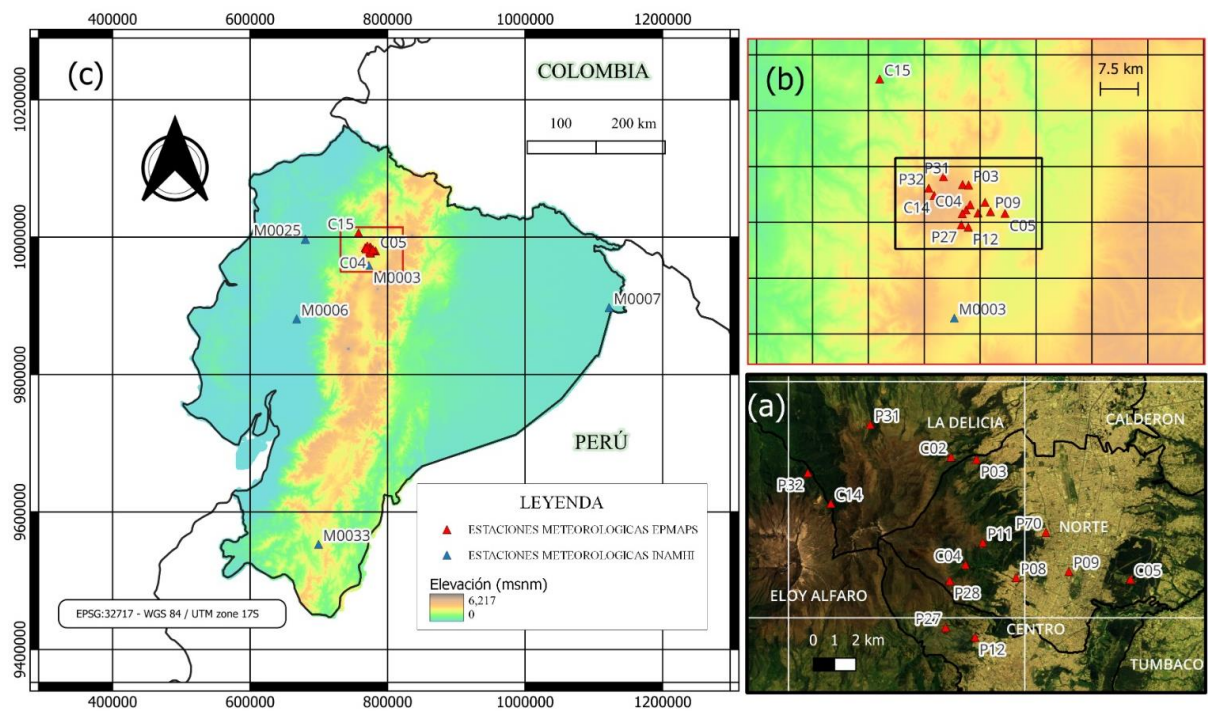


Figura 3. Mapa geográfico de las estaciones meteorológicas utilizadas en el Ecuador.

(a) Mapa de las estaciones meteorológicas de la EPMAPS dentro del Distrito Metropolitano de Quito con una grilla representativa del tamaño del píxel (aproximadamente 10 km x 10 km), (b) mapa referencial de las estaciones meteorológicas de la EPMAPS con una grilla representativa del tamaño del píxel (aproximadamente 10 km x 10 km), (c) mapa topográfico del Ecuador con las estaciones meteorológicas del EPMAPS y el INAMHI.

Elaboración: propia

El Ecuador es un país andino, amazónico y costero, un lugar lleno de discordancias humanas, físicas, geológicas, climáticas y evidentemente hidrológicas. En este país se encuentra yuxtapuesto a breves distancias, terrenos turbados y llanuras, zonas cálidas y

zonas frías, regiones húmedas y secas, en donde el agua se manifiesta de maneras variables. Uno de estas casos variables es la distribución de las lluvias, que dependen de componentes planetarios que rigen las situaciones climáticas, entre los cuales se encuentra la circulación atmosférica general o incluso la existencia de corrientes oceánicas (Pourrut, 1995).

Una segunda interdependencia es el contexto físico local, las condiciones pluviométricas en Ecuador están gobernadas por leyes climáticas propias de las zonas de baja latitud (desplazamiento periódico del Frente Intertropical, FIT o también llamado Zona de Convergencia Intertropical, ZCIT, regímenes de los alisios, etc.) y dependientes de corrientes oceánicas (desplazamientos de la corriente fría de Humboldt, alteración climática ocasionada por el fenómeno de El Niño), sin olvidar el efecto de la cordillera andina que tiene un papel predominante en el origen, el aislamiento y la repartición de masas de aire cuyas características afectan en la formación de las precipitaciones (Pourrut, 1995).

Otros factores importantes son el Sistema Monzónico Sudamericano (SAMS) es la característica climática más importante del continente. Proporciona recursos hídricos a millones de personas que viven en áreas rurales y urbanas, y es un control dominante tanto en la agricultura como en la generación de energía hidroeléctrica. SAMS exhibe variaciones en una amplia gama de escalas, que además dependen de complejas interacciones multisistema, retroalimentación y teleconexiones (De Carvalho & Cavalcanti, 2016). Así como, la desviación de la humedad del aire hacia el sureste da ascenso al jet de bajo nivel sudamericano (SALLJ) que transporta la humedad del aire a lo largo de los Andes orientales hacia la cuenca del Plata (Boers et al., 2013).

Para nuestro estudio se realizará una comparación entre tres regiones, cada una cuenta con características bien marcadas, propias y únicas.

Tabla 9. Regiones del Ecuador con su dependencia regidora de precipitación

Región	Elevación (m s. n. m.)	Regidor de clima	Régimen de lluvia
Costa	0 - 1000	ITCZ	Húmeda (diciembre a mayo), seca (junio a noviembre)
Sierra	> 1500	Elevación, ITCZ	Húmeda (diciembre a mayo), seca (junio a noviembre)

Amazonía	0 - 500	ITCZ, SAMS, vientos alisios	Estacionalidad débil (más seco junio, julio, agosto)
----------	---------	-----------------------------	--

Fuente:(Manz et al., 2017)

La variabilidad espacial de las precipitaciones es la principal fuente de incertidumbre en la hidrología de las cuencas (Buytaert et al., 2006). La inestabilidad interanual de las precipitaciones está modulada en parte por las extravagancias de la temperatura de la superficie marina de los océanos Pacífico y Atlántico tropicales (Vuille et al., 2000). Entre ellas, El Niño Oscilación del Sur (ENOS) produce anomalías positivas de precipitación que son responsables de varios impactos socioeconómicos en las poblaciones andinas ecuatoriales, como la de Quito (Vuille et al., 2000). Además, la Oscilación Madden-Julian (MJO) impulsa la variabilidad de las precipitaciones intraestacionales en Ecuador (Recalde-Coronel et al., 2020)

Por otro lado, en la parte oriental de los Andes (DMQ) se presenta un régimen de precipitación unimodal distinto con un pico en el austral invierno, asociado a la advección de vapor de agua desde la cuenca amazónica por el incremento de los vientos del este durante esta estación (Ruiz-Hernández et al., 2021). Además, en los valles interandinos se presenta un ciclo estacional bimodal, con máximos en marzo-mayo y octubre-noviembre, que se identifica por condiciones secas durante el invierno austral relacionadas con una fuerte subsidencia atmosférica (Campozano et al., 2016).

2.1 Metodología

En el presente acápite se especifica la metodología utilizada para determinar las curvas IDF utilizando IMERG V06.

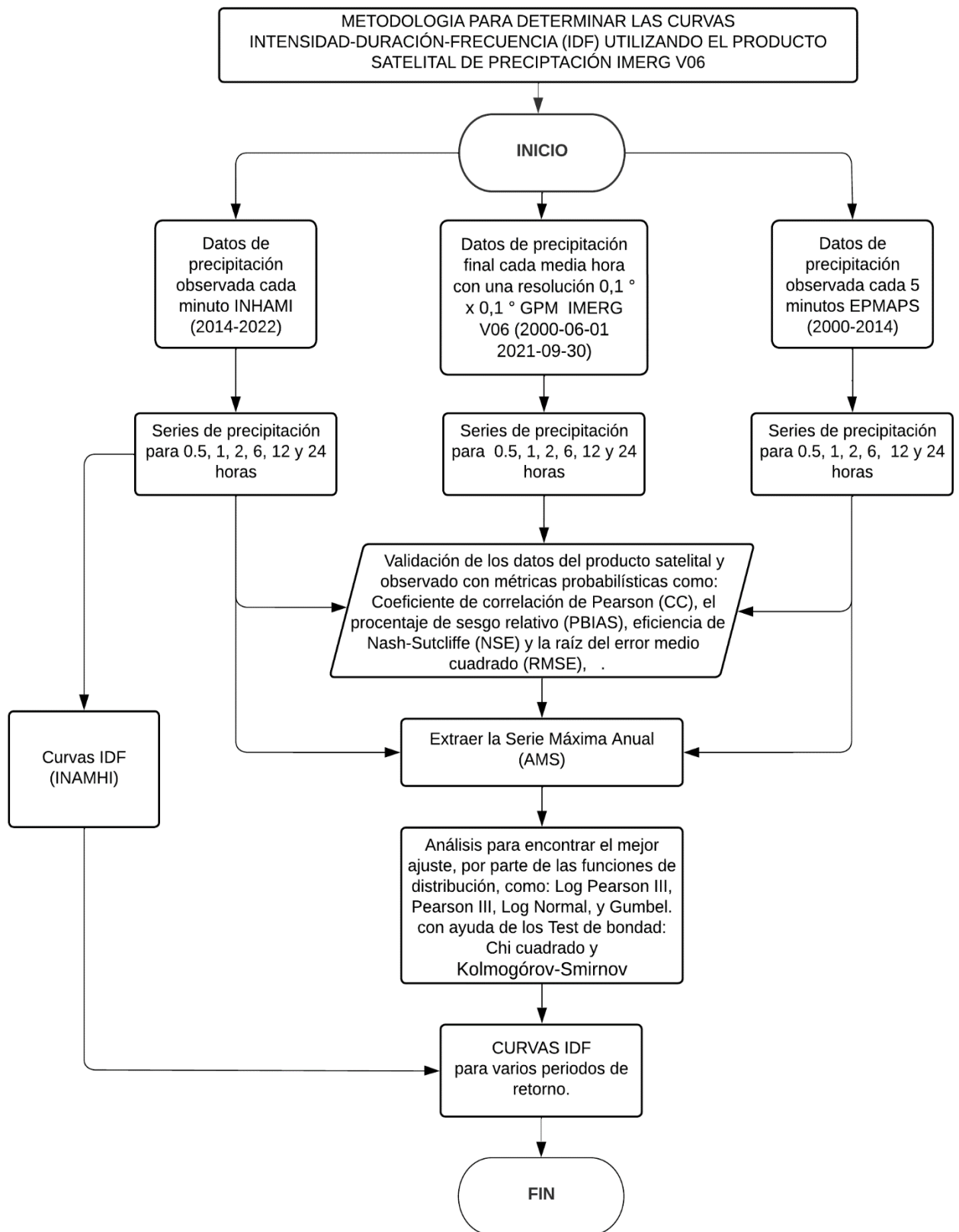


Figura 4. Diagrama de flujo de la metodología empleada

Fuente: Autoría propia

2.1.1 Recopilación de información

Precipitación Satelital

Los datos de IMERG V06 se pueden descargar libremente de la página GES DISC administrada por la NASA. En la interfaz de búsqueda perteneciente a este portal se debe colocar IMERG V06. Hay varias versiones relacionadas con IMERG V06. Para este estudio se utilizó los datos en su versión final porque en esta presentación se tienen correcciones de las observaciones de campo calibrando los datos de precipitación de los satélites GPM. Además, se eligió los datos con un intervalo de 30 minutos con una resolución espacial de 0.1° aproximadamente 10 km. Para empezar con la descarga se debe insertar las siguientes características:

- El método de la descarga el cual es a través de un archivo de texto en OPeNDAP que contiene links de los datos geoespaciales.
- Las coordenadas decimales de la ubicación de nuestra zona de estudio, es recomendable seleccionar un tanto más de la zona necesaria debido a la influencia de los pixeles.
- Las fechas de los datos, que será el periodo en el que exista información disponible.
- Formato de salida de datos en NetCDF compatible con el software R.
- La variable es precipitationCal recomendada para uso general e investigaciones.

Tabla 10. Lista de las variables de los datos de salida IMERG V06 de media hora.

Datos Media Hora (Temprano, Tardía, Final)	
precipitationCal	Estimación de precipitación multisatélite con calibración de medidor (recomendado para uso general)
precipitationUncal	Estimación de precipitación multisatélite
Random Error	Error aleatorio para la precipitación multisatélite calibrada por pluviómetro
HQprecipitation	Estimación de precipitación combinada solo de microondas
HQprecipSource	Identificador de fuente de satélite de microondas
HQobservationTime	Tiempo de observación del satélite de microondas
probabilityLiquidPrecipitation	Probabilidad de fase de precipitación líquida
PrecipitationQualityIndex	Índice de calidad para el campo precipitationCal
IRkalmanFilterWeight	Ponderación de la precipitación solo IR en relación con la precipitación solo de microondas fusionada transformada
Precipitation	Estimación de precipitación solo IR

Fuente: (Huffman et al., 2017)

A continuación, con un software de descarga de datos desde servidores web se obtiene la información, como se indica en el Anexo I. En el procesamiento de los datos descargados se debe considerar que los archivos entregan información que está en un horario (UTC 00:00) diferente a la de nuestro país que es (UTC-05:00 QUITO/BOGOTA), y las unidades de datos vienen dadas en mm/h, con dimensión de intensidad de precipitación.

$$i = \frac{hp}{t}$$

Ec. 32

Donde:

i = Intensidad de precipitación (mm/h)

hp = Altura de precipitación (mm)

t = Tiempo (h)

Precipitación Observada

La precipitación observada en un lugar específico significa, tomar datos de la medida de la lluvia en un intervalo determinado a través de una estación pluviométrica. Las estaciones meteorológicas usadas para obtener los datos de precipitación observada son administradas por dos instituciones públicas de Ecuador, la primera es el Instituto Nacional Meteorología e Hidrología (INAMHI), y la segunda es la Empresa Pública Metropolitana de Agua Potable y Saneamiento (EPMAPS).

INAMHI

El Instituto Nacional de Meteorología e Hidrología (INAMHI) es un centro científico que recoge y divulga los datos históricos, actuales y futuros acerca del tiempo, clima y recursos hidrometeorológicos del Ecuador. Esta entidad es parte de la Organización de las Naciones Unidas para la Meteorología, conocida como OMM, la cual le obliga a seguir estándares internacionales en busca de mejorar sus predicciones del tiempo, a través del uso de tecnología, normas y otros tipos de regulaciones (Apolo Cumbicos & Taco Taco, 2022).

El INAMHI dispone de 260 estaciones, sin embargo, para este estudio se solicitó información de cinco estaciones: Izobamba (M003), Pichilingue (M006), Nuevo Rocafuerte

(M007), La Concordia (M025) y La Argelia (M033). Las fechas y la ubicación de las estaciones se presenta en la siguiente tabla:

Tabla 11. Periodo de registro de datos de las estaciones del INAMHI

Código	Nombre	Provincia	Periodo De Registro De Datos	
			Fecha Inicio	Fecha Final
M0003	Izobamba	Pichincha	7/11/2014	31/12/2022
M0006	Pichilingue	Los Ríos	22/05/2013	17/03/2018
M0007	Nuevo Rocafuerte	Orellana	13/05/2015	27/06/2016
M0025	La Concordia	Santo Domingo de los Tsáchilas	19/07/2014	17/03/2018
M0033	La Argelia-Loja	Loja	22/05/2013	17/03/2018

Fuente: Elaboración con información de (INAMHI ,2022)

Con la observación de que para la estación M0025 La Concordia en su periodo de registro, no se contaba con el año 2016.

EPMAPS

Es una empresa pública enfocada a mejorar la calidad de servicios que ofrece a la ciudadanía del Distrito Metropolitano de Quito (DMQ), maneja la Red de Monitoreo Hidrometeorológico de la EPMAPS, esta red tiene estaciones ubicadas en sitios estratégicos dentro y fuera de Quito (EPMAPS & FONAG, 2021).

La EPMAPS dispone de 89 estaciones, para este estudio se solicitó la información de quince estaciones ubicadas en el DMQ. Las fechas de los datos de las estaciones se presenta en el siguiente cuadro:

Tabla 12. Periodo de registro de datos de las estaciones del EPMAPS

Código	Nombre	Periodo De Registro De Datos	
		Fecha Inicio	Fecha Final
C02	Rumihurco	2/5/2000	7/1/2014
C04	Rumipamba	5/10/2000	9/6/2014
C05	Bellavista	28/5/2000	14/7/2012
C14	Mindo Captación	2/1/2006	8/11/2016
C15	Nanegalito	2/2/2012	7/8/2022
P03	Rumihurco Machángara	24/5/2000	29/9/2012
P08	Rumipamba Bodegas	6/12/2000	17/5/2012
P09	Iñaquito INAMHI	30/5/2000	22/9/2012
P11	Antenas	13/6/2000	12/9/2012
P12	Toctiuco	1/7/2000	11/10/2012
P27	San Francisco	23/7/2003	27/12/2014

P28	Cruz Loma	17/2/2003	14/12/2013
P31	Pichán	26/11/2002	10/11/2015
P32	Mindo Bajo	26/11/2002	12/1/2006
P70	CC El Bosque	14/12/2015	23/12/2023

Fuente: Elaboración con información de (EPMAPS, 2023)

En total son 20 estaciones, a continuación, se presentará un cuadro de resumen del porcentaje de vacíos de los datos, las coordenadas geográficas y la altura que describe la ubicación de las estaciones:

Tabla 13. Características de las estaciones utilizadas

N°	CÓDIGO	NOMBRE	% VACIOS	LONGITUD (°)	LATITUD (°)	ALTURA (m s. n. m.)
1	M0003	Izobamba	38.85%	-78.555	-0.366	3058
2	M0006	Pichilingue	12.80%	-79.462	-1.100	120
3	M0007	Nuevo Rocafuerte	47.53%	-75.403	-0.917	185
4	M0025	La Concordia	53.31%	-79.380	-0.025	379
5	M0033	La Argelia-Loja	10.96%	-79.203	-4.038	2160
6	C02	Rumihurco	0.03%	-78.537	-0.130	3586
7	C04	Rumipamba	0.42%	-78.531	-0.175	3355
8	C05	Bellavista	0.20%	-78.462	-0.182	2960
9	C14	Mindo Captación	2.21%	-78.588	-0.149	3600
10	C15	Nanegalito	0.39%	-78.685	0.060	1767
11	P03	Rumihurco Machángara	0.38%	-78.527	-0.131	3245
12	P08	Rumipamba Bodegas	0.00%	-78.510	-0.181	3005
13	P09	Iñaquito INAMHI	0.32%	-78.488	-0.178	2804
14	P11	Antenas	0.00%	-78.524	-0.166	3800
15	P12	Toctiuco	0.06%	-78.527	-0.206	3232
16	P27	San Francisco	0.71%	-78.540	-0.202	3551
17	P28	Cruz Loma	0.10%	-78.538	-0.182	4012
18	P31	Pichán	0.00%	-78.572	-0.116	3516
19	P32	Mindo Bajo	0.00%	-78.598	-0.136	3330
20	P70	CC El Bosque	0.00%	-78.497	-0.162	2903

Fuente: Elaboración con información de (INAMHI ,2022) y (EPMAPS, 2023)

2.1.2 Lectura de datos satelitales

Descargado los datos de IMERG V06, lo siguiente es extraer la información que contienen. El primer paso es la lectura de estos con ayuda del software RStudio, pero antes debemos entender que el formato NetCDF es una superposición de datos, que almacenan información multidimensional. Este formato es útil cuando se trata de acoger objetos que contengan diferentes tipos de datos en un entorno de red heterogéneo y para escribir

software de aplicación que no depende de formatos específicos. Es decir, una superposición de capas temáticas que pueden ser representadas por su momento temporal o alguna característica diferente (Rew & Davis, 1990).

En nuestro caso los datos NC4 (NetCDF versión 4) contiene varios píxeles de tamaño (10 x10 km) donde se guarda la información numérica sobre la precipitación expresada en mm/h (precipitacionCal). El lenguaje R tiene un sin número de librerías en varios tópicos, donde se guardan millones de funciones disponibles, solamente hace falta cargarlas para poder utilizarlas, esto nos simplifica un poco el trabajo. Para extraer la información de esta especie de imagen raster, es necesario dar una característica propia de cada píxel para llegar a los datos adicionales contenidos en el mismo.

En este caso, se ha ingresado las coordenadas de las estaciones pluviométricas para acceder a los datos, como se ve en el Anexo II. Se creó un objeto data frame (tabla en R) que contiene el nombre de todas las estaciones con sus respectivas coordenadas decimales (latitud y longitud), para poder sacar los datos de IMERG V06 de una sola ejecución del script, formando así un gran data frame que contenga todos los datos disponibles de cada una de las estaciones. Los datos se encuentran en unidades de (mm/h), para obtener magnitudes de precipitación se tiene que multiplicar por el tiempo de la resolución temporal (30 minutos) pero transformado a horas para poder obtener (mm).

En el caso de la información proporcionada por la EPMAPS, se recibió los datos observados por las estaciones meteorológicas en archivos Excel de formato .csv por cada una de las estaciones, este documento contenía datos medidos cada 5 minutos de los años en los que existía registro. Por otra parte, en el caso de la información proporcionada por el INAMHI, se recibió los datos observados por las estaciones en archivos Excel de formato .csv, pero cada estación tenía por cada año un archivo con los datos medidos cada minuto. Por lo que, se complicó la lectura de estos, ya que la información para cada estación debe seguir una secuencia lógica de tiempo en una sola columna, para facilitar el tratamiento más adelante en el análisis.

2.1.3 Series de datos

Toda la información se encuentra disponible en el lenguaje de tratamiento de datos R, el siguiente paso es agregar los datos a diferentes series de tiempo: 30 minutos, 1 hora, 2 horas, 6 horas, 12 horas y 24 horas. Una vez que los datos IMERG V06 se han transformado los datos a altura de precipitación, se deben pasar a la zona horaria de Ecuador (UTC -05:00) ya que los datos están en (UTC 00:00), para esto solo se deben restar las 5 horas transformada a segundos (18000s) a cada uno. Para realizar la

agregación temporal se tiene la primera serie de 30 minutos del IMERG V06, a partir de esta serie se podrá obtener las demás con un proceso de suma según los intervalos de tiempo requeridos. Agregación es sumar sucesivamente los valores disponibles de precipitación de una estación en un periodo determinado (1 hora, 2 horas, 6 horas, 12 horas y 24 horas). Se creó un script de R con el proceso de agregación el cual generará las series de tiempo ver el Anexo III.

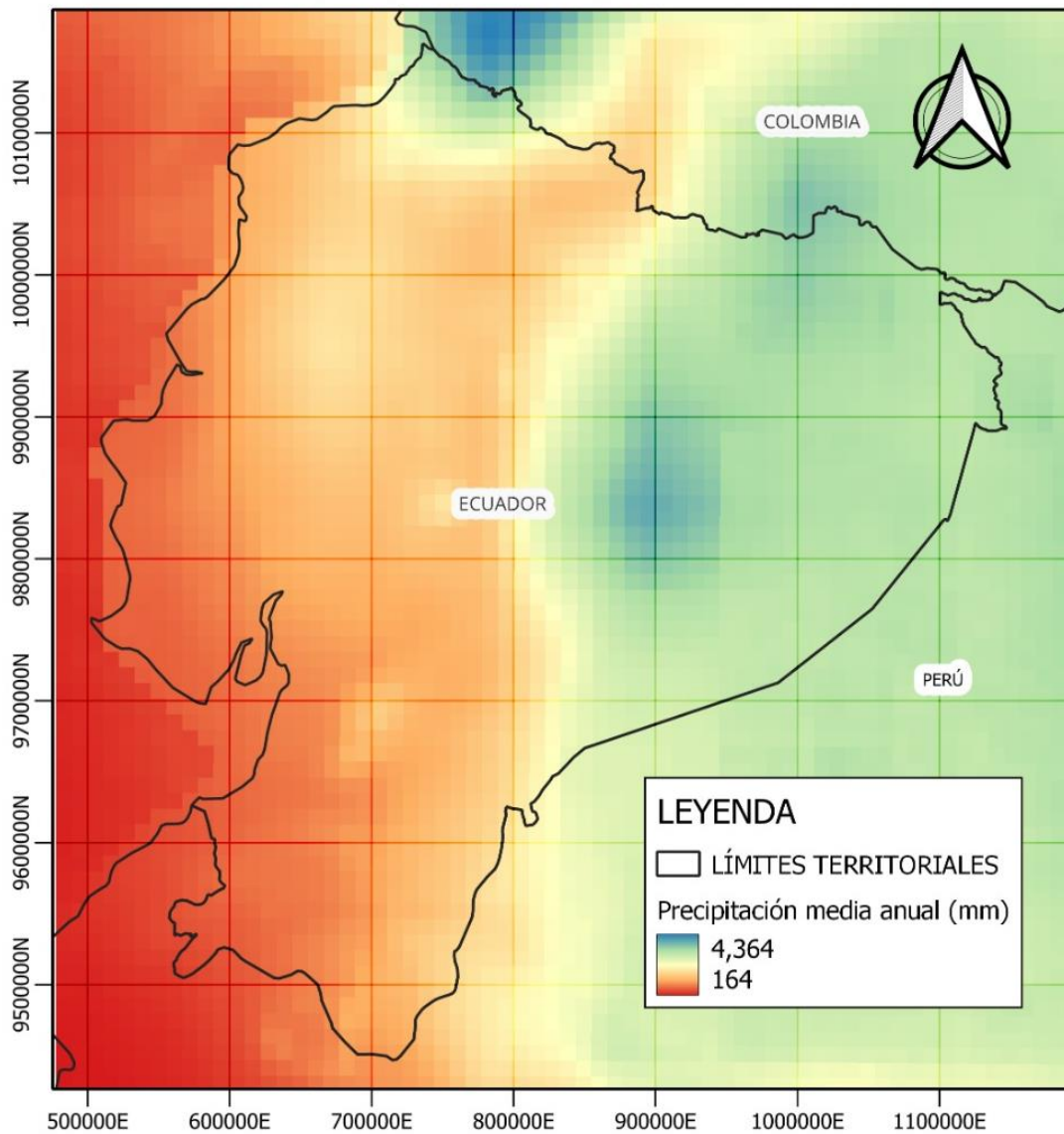


Figura 5. Serie de Precipitación Media Anual (2000-2021)

Fuente:(Parsons et al., 2010)

Elaboración: propia

La toma de datos de precipitación observada del INAMHI y EPMAPS se realiza con estaciones calibradas a la zona horaria ecuatoriana, por eso no se necesitan cambios adicionales como en el IMERG V06. Por otro lado, se añade una serie adicional de 30 minutos ya que los datos vienen dados del INAMHI cada minuto Anexo III y EPMAPS cada cinco minutos el Anexo III. De manera similar al IMERG V06 se realizará la agregación a partir de la serie temporal de 30 minutos para todos los demás tiempos.

Se debe de tener precaución que el formato para las fechas este en POSIXct y para los valores de precipitación este en formato numeric para que el programa pueda trabajar sin ningún problema. En los datos de una estación meteorológica se puede encontrar valores que no se captan, o que no se procesan por cualquier razón que fuese. En una serie de tiempo de 1 minuto puede haber valores faltante y saltos de tiempo, y lo mismo ocurre con la de 5 minutos, pero el programa no lee, solo lee que hay que hacer un índice de fechas cada minuto y rellenando las fechas vacías con valores sin datos (NA). Esto afecta el proceso de agregación debido que el programa al sumar datos existentes con NA el resultado será NA, disminuyendo así en cada agregación la cantidad de datos válidos para el análisis.

La estación M0025 del INAMHI presentó datos equívocos debido a la imposibilidad de tener valores medidos tan grandes en precipitaciones de 1 minuto, ni en eventos tan extremos se puede llegar a medir (300 mm) de precipitación en un solo minuto. Para resolver este problema se realizó un control de calidad que elimine estos valores, (Shen et al., 2010) propone el control con verificación extrema el cual consiste hacer una comparación del valor de precipitación de 30 minutos con el valor máximo de la serie temporal de precipitación diaria. A través del script del Anexo III con un condicional se pudo eliminar los valores erróneos de la serie, resultando en una disminución de la cantidad de valores para esta estación, pero los valores que quedaron son correctos.

2.1.4 Validación IMERG

Los datos compilados de las estaciones meteorológicas en periodos de tiempo extensos pueden afectar o presentar perturbaciones, por la relocalización, mal uso o deterioro de los equipos, condiciones climáticas extremas o errores de medición (Villón, 2002). Por eso, antes de construir las curvas, es necesario realizar un proceso de comprobación de validez con técnicas estadísticas para garantizar que los datos sean limpios, precisos y, por ende, utilizables. En el presente estudio se realizó este proceso a través del programa R ver los códigos en el Anexo IV.

Este proceso se basa en realizar una comparación de los datos observados (INAMHI, EPMAPS) con los datos obtenidos por el producto satelital IMERG V06. Los procedimientos utilizados es este apartado son: (CC) Coeficiente de Correlación en el cual ocuparemos el método de Pearson; (RMSE) error medio cuadrático; (PBIAS) porcentaje BIAS y (NSE) la eficiencia Nash-Sutcliffe. Los datos se unirán en una gran tabla por cada duración donde se encuentre señalado el índice de las columnas con el nombre de la estación con su respectiva procedencia (INAMHI, EPMAPS) y el índice de filas será la fecha de la toma de datos que dependerá de la variación del tiempo de la serie (30 minutos, 1 hora, 2 horas, 6 horas, 12 horas, 24 horas), todo lo demás se llenará de valores numéricos de precipitación en mm y valores vacíos NA.

Entonces, habrá 6 grandes tablas una por cada duración de tiempo, antes de unir las tablas entre observadas y satelitales se debe de verificar que las agregaciones se hayan dado en el mismo tiempo, por ejemplo: para la serie de 12 horas la agregación para IMERG V06 se dio a las 7 am y 19 pm, entonces para la precipitación observada también se debe de realizar la agregación a estas horas, por lo que solamente se puede unir si son datos equiparables es decir que van a estar en la misma unidad y en el mismo instante.

Unidos los datos se debe de verificar que coincida las agregaciones realizadas en el programa R con la agregación manual que se puede realizar con los archivos Excel de las estaciones observadas, si estos dos valores son iguales en valor y en tiempo, se puede decir que la agregación se realizó con éxito y se puede seguir al siguiente paso. La validación es el siguiente paso, el cual inicia cargando las librerías que contenga las funciones (COR, PBIAS, RMSE, NSE), luego en dos tablas una con los valores de las estaciones observadas y la otra de los valores satelitales.

A través de una iteración se hará correr las funciones por todas las columnas de la tabla, solo se tiene que consultar el orden en el que van los datos en la función para que el código funcione, al realizar la consulta resulta que el primer lugar es para los datos satelitales y segundo lugar es para los datos observados. Este orden también ayudará luego a inferir algunas hipótesis conforme se analice los resultados.

Después del análisis se graficaron los resultados, para ver de manera clara el comportamiento de cada una de las estaciones, en el eje x estarán el nombre de las estaciones y en el eje y se encuentran los resultados de cada uno de los coeficientes, el Coeficiente de Pearson y el NSE no tienen unidades, el PBIAS se mide en porcentaje, y el RMSE tiene unidades de los valores con los que estamos trabajando en este caso de

precipitación en mm. La duración también es parte de este gráfico por lo que cada color representa una duración por lo que por cada estación habrá 6 barras.

En este script también se ha revisado la calidad de los datos observados, otra característica importante es controlar la cantidad de datos, por lo que se determinó el % de valores vacíos de cada una de las estaciones. Las estaciones de la EPMAPS tienen un insignificante porcentaje no mayor al 3%, por otro a parte los datos del INAMHI presentan considerables porcentajes de vacíos con una media del 32.69% de vacíos en sus datos, lo que dificultará la comparación y su análisis.

2.1.5 Series máximas anuales

Las series máximas anuales de precipitación se obtiene en todas las duraciones 30 minutos, 1 hora, 2 horas, 6 horas, 12 horas y 24 horas por cada estación, considerando que se tienen dos tipos de datos entre observados y satelitales. En total se obtendrán 240 series máximas anuales, 12 por cada estación, 6 para precipitación satelital y 6 para precipitación observada.

Los resultados obtenidos del programa se exportarán a una tabla en Excel de formato .csv, para después ingresarlos nuevamente al programa R y graficar la serie máxima anual de la precipitación observada, con la precipitación satelital realizando una comparación por píxeles, es decir la mayoría de las estaciones se encuentra sola en un píxel en este caso se graficará los resultados de la estación satelital y la estación meteorológica. Pero hay casos en los que hay dos, tres, cuatro y nueve estaciones en un mismo píxel, ahí se precede a graficar los dos, tres, cuatro y nueve resultados y solamente una estación aleatoria del IMERG V06.

2.1.6 Ajuste de frecuencia

El siguiente paso para la construcción de las curvas IDF se efectúa por medio del uso de una metodología de cálculo que tiene en cuenta la necesidad de un ajuste a los valores de precipitación a una Función de Distribución de Probabilidad (CDF). Como se obtuvieron 240 series máximas anuales se realizará 240 ajustes con cada una de las funciones como: Log Normal, Gumbel, Pearson III y Log Pearson III. No existe una fórmula mágica que diga que la función se ajusta mejor a nuestros datos, por eso hay que probar para cada caso que es la que mejor calza. Una diferencia trascendental entre las funciones es la facilidad de su aplicación, ya que, por ejemplo, las funciones de Gumbel y Log normal cuenta con 2 parámetros a ajustar, mientras Pearson III y Log Pearson III tiene 3 parámetros (Tapia et al., 2012).

La función de distribución de Gumbel es la más usada para este tipo de estudios, pero ha recibido críticas porque podría reemplazarla por la distribución de valor extremo tipo 2 (Soro et al., 2010). Los mismos autores al ocupar las distribuciones Valor extremo Tipo I o Gumbel y Log normal para 13 estaciones del área tropical de África, descubrieron que las funciones mejor se ajustaron a estos datos fueron Gumbel y Log normal, y solo un 9 % de las series se ajustaron mejor a otras.

Para (Lau & Behrangi, 2022), en su estudio en Austria y Arizona, algunos datos podrían ajustarse favorablemente con cualquiera de las 4 CDF (valor $p > 0,05$). Con más pruebas, la función de Valores Extremos Generalizada (GEV) con la Estimación de Máxima Verosimilitud (MLE) dan el ajuste más apropiado (valor p máximo = 0,89) para todos los conjuntos de datos de ambas regiones. Es notorio que se genera una dependencia entre elegir la distribución correcta y la duración del registro, o sea, que estos estudios arrojaran mejores resultados cuando haya más datos. En (Kyaw et al., 2022), la función de mejor ajuste es Gumbel con MLE (valor p máximo = 0,87), y por último Pearson 3 que consiguió el peor ajuste (valor p máximo = 0,81) para todos los conjuntos de datos en ambas zonas.

En Colombia, (Rodríguez-Sandoval et al., 2023), ocuparon la distribución de probabilidad de Gumbel a razón de que el Instituto de Hidrología, Meteorología y Estudios Ambientales (IDEAM) usó esta distribución para elaborar las curvas IDF, y ha sido reportada como la que mejor se ajusta a las intensidades máximas anuales de los datos IMERG. Otro caso colombiano según (Acosta Castellanos & Sierra Aponte, 2013), utiliza Gumbel debido a la flexibilidad para su uso. Es la más idónea para ajustar precipitaciones máximas en diferentes periodos de retorno para la región además de su rapidez y facilidad de cálculo. Y solo cuando se tiene continuidad en el registro histórico es factible ocupar otro tipo de función como log Pearson tipo III. En nuestro caso, el INAMHI realiza un análisis de todas las funciones de distribución y luego selecciona la que mejor se ajusta. Por esta razón se ocupará este método analizar todas las CDF para luego seleccionar la que mejor se ajusta.

Con las series máximas anuales ya disponibles, lo siguiente será realizar el análisis de la serie para la función Gumbel, Log Normal, Pearson III y Log Pearson III, para esto en R es necesario cargar la librería hydroGOF que contiene la mayoría de las funciones que se necesita para generar el script. El script se desarrolla escribiendo las fórmulas de cada una de las funciones de distribución con sus respectivos parámetros ver Anexo V. Anteriormente, ya se mostró que Gumbel y Log normal debido a que tenían dos parámetros tenían una ventaja en cuanto a facilidad y rapidez que también se nota al momento de transcribir sus fórmulas al código, por otro lado, Pearson III y Log Pearson III cuenta con

un grado más de dificultad debido a sus tres parámetros de escala forma y ubicación. Todas las funciones calcularon sus parámetros por el método de los momentos (MME).

2.1.7 Test de bondad de ajuste

La prueba sirve para escoger la función que mejor se ajusta, se ha realizado dos pruebas Chi-cuadrado y Kolmogórov-Smirnov. Se inicia cargando las librerías pertenecientes a las funciones de las pruebas, en este caso será la misma que para las CDF librería hydroGOF. Al correr el programa por cada una de las series máximas anuales se podrá obtener resultados de cada una de las estaciones y con esto se podrá realizar las pruebas, simplemente se tiene que añadir los resultados de las CDF a las funciones chisq y ks.test. Ahora estos resultados se exportarán a una tabla de Excel para realizar un tratamiento manual y elegir la función que mejor se ajuste. Se ha decidido otorgar más relevancia a la prueba de Kolmogórov-Smirnov para elegir la función idónea, y si se tiene valores iguales en esta prueba se elegirá por la función que más veces se haya ajustado mejor anteriormente en las otras estaciones. En la función Chi cuadrado si se tiene menos de 10 datos el programa no corría debido a que es necesario esta cantidad de datos para realizar la prueba de ajuste, por lo que ciertas estaciones principalmente las del INAMHI no pudieron ser analizadas por esta prueba. Por otro lado, la prueba de Kolmogórov-Smirnov se ejecutaba en todas las sin ningún problema, aunque haya carencia de datos Anexo V.

2.1.8 Periodo de retorno

El periodo de retorno que ocupamos será 2 y 50, debido a que hemos ocupado una serie máxima anual de 20 años y la realización de predicciones solamente es factible para un máximo del triple de los datos que tenemos en la muestra, en este caso el triple de 20 años vendría a ser 60 años. Con esto podemos obtener las precipitaciones para un periodo de retorno de 2 y 50 años a diferentes duraciones 30 minutos, 60 minutos (1 hora), 120 minutos (2 horas), 360 minutos (6 horas), 720 minutos (12 horas), y 1440 minutos (24 horas). En R se cargará la librería HydroGOF y con ayuda de la probabilidad, frecuencia y periodo se puede obtener la estimación de precipitación en mm. Estos resultados los exportaremos en una tabla .csv de Excel para realizar un tratamiento manual, ya que debemos de solamente elegir los valores de precipitación de las funciones que mejor se ajusten, los demás se desechan Anexo VII.

2.1.9 Curvas IDF

Los resultados anteriores se encuentran en tablas de Excel para poder ingresarlos a R debemos de cambiar el formato a .csv delimitado por comas UTF-8. En R se carga la librería de grafico ggplot2, las gráficas se realizarán nuevamente en comparación por

pixeles y la estación o estaciones que están ubicadas en cada píxel. Las estaciones del INAMHI M0003, M0006, M0007, M0025 y M0033 cuentan con un registro de datos muy pobre, lo cual dificulta el análisis anterior. Por esta razón, se decidió ocupar los valores entregados por el INAMHI de la serie máxima anual, la mejor distribución, y los valores de las intensidades para su respectivo periodo de retorno con estimaciones de precipitación. Posterior a realizar el análisis se tiene valores de precipitación en unidades de mm, pero las curvas IDF son Intensidad-Duración-Frecuencia entonces debemos de cambiar la precipitación mm por Intensidad mm/h, simplemente ocupando la ecuación (Ecu. 32) debemos dividir los resultados de precipitación estimada por cada duración para conseguir la intensidad estimada.

Luego, se graficará las curvas IDF observadas con las curvas IDF satelitales, nuevamente caracterizada por la comparación entre pixeles a los que pertenece cada estación. En el eje X se graficará la duración de 30 minutos (0.5 horas), 60 minutos (1 hora), 120 minutos (2 horas), 360 minutos (6 horas), 720 minutos (12 horas), y 1440 minutos (24 horas). En el eje Y se graficará los valores de intensidad, el tipo de línea y el grosor de esta diferenciará el periodo de retorno (2 y 50 años) Frecuencia y por último el color de la línea de la filtrará si la estación pertenece a una curva IDF en base a precipitación observada o en base al producto IMERG V06 Anexo VII.

3 RESULTADOS, CONCLUSIONES Y RECOMENDACIONES

3.1 Resultados

3.1.1 Series temporales

Las siguientes gráficas muestran las precipitaciones de las estaciones observadas comparadas con las precipitaciones satelitales, las imágenes están divididas por pixeles de la imagen raster de los datos descargados de IMERG V06. La línea azul representa los datos satelitales y la roja es la precipitación observada, cuando exista más de una estación en un píxel se graficará la otra serie con otro color diferente de la paleta de colores roja. En las abscisas se ubicará el periodo de registro de los datos (2000-2021) y en las ordenadas se ubicará las precipitaciones mensuales medidas en mm. La serie de 30 minutos se tomó de ejemplo para obtener la agregación mensual debido a que es la muestra inicial, la cual sirvió para obtener las agregaciones.

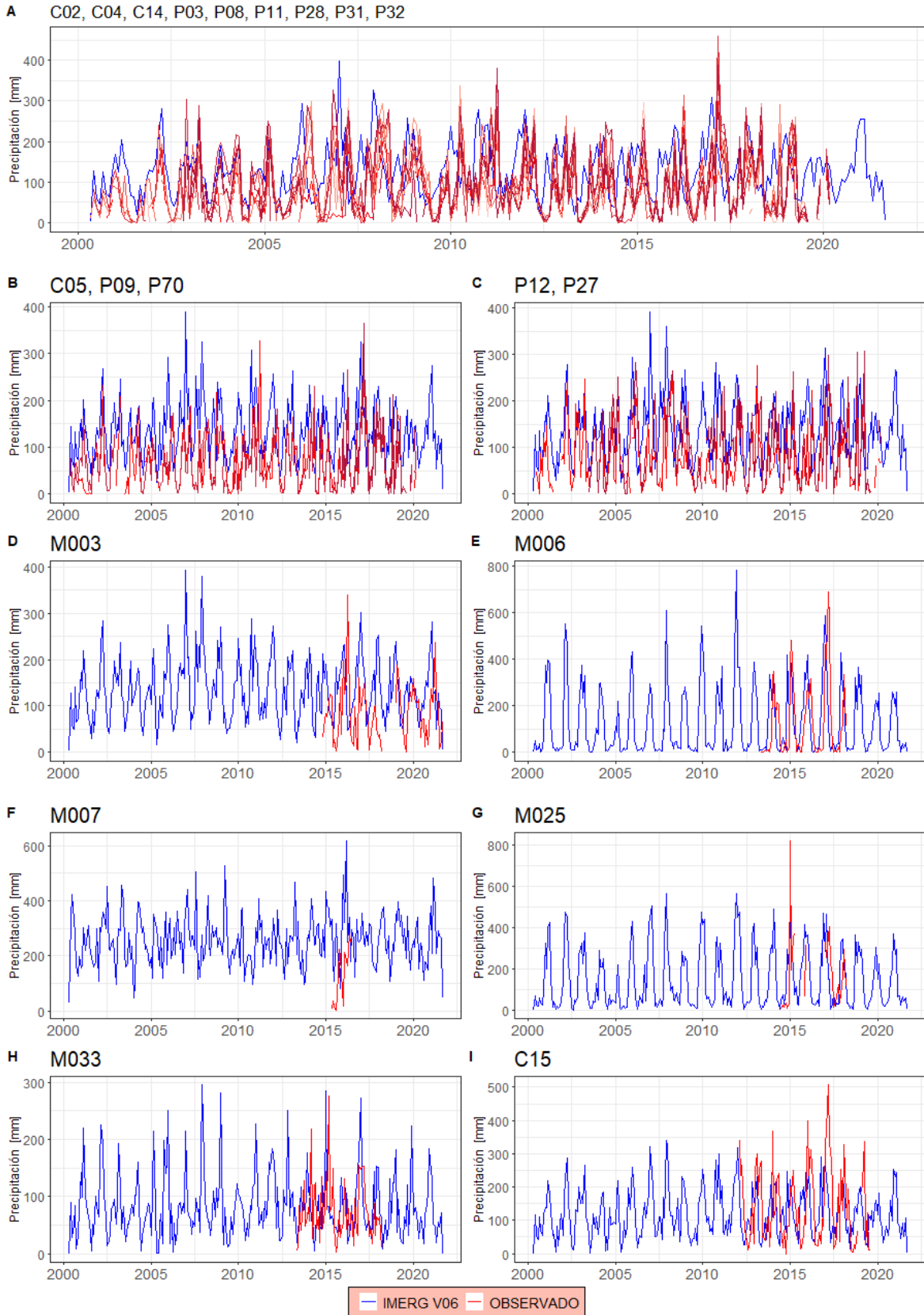


Figura 6. Series temporales de precipitación mensual IMERG V06 vs Observadas.

Fuente: (Huffman et al., 2019)

Elaboración: propia

La (Figura 6A) pertenece al píxel que cuenta con 9 estaciones (C02, C04, C14, P03, P08, P11, P28, P31, P32) ubicado en nororiental del DMQ. Es notorio que la serie temporal de IMERG V06 se encuentra por encima de las series observadas de la EPMAPS, es decir que esta sobreestimada. Las mediciones observadas solamente sobrepasan a las satelitales cuando existen picos, es decir lluvias importantes tomadas por las estaciones meteorológicas. Las estaciones P31, C14, P32 tienen elevaciones mayores a 3500 m s. n. m. todas las demás estaciones se encuentran en una especie de planicie, sin embargo no hay un desfase importante que nos muestre alguna anomalía.

La (Figura 6B) pertenece al píxel que cuenta con 3 estaciones (C05, P09, P70) ubicadas en la parte occidental del DMQ, ocurre exactamente lo mismo los datos satelitales se encuentran por encima de los datos observados, es decir que se sobreestima los datos del IMERG V06. Existen dos picos (2011-2017) por parte de EPMAPS que están seguidos de picos por parte de IMERG V06, creando una relación.

La (Figura 6C) pertenece al píxel que cuenta con 2 estaciones (P12, P27), en su gran mayoría los datos observados se encuentran encima de los datos satelitales, es decir que están sobreestimados los datos del satélite. Existen cuatro picos (2006-2007-2010-2011) por parte de EPMAPS, sin embargo no existe relación por parte IMERG V06. La (Figura 6I) pertenece a la estación C15, la cual es la única estación sola y diferente de la EPMAPS donde se puede notar una subestimación del producto IMERG V06.

Los gráficos con estaciones del INAMHI (M0003, M0006, M0007, M0025, M0033), tienen escasos de datos por lo que se podrá hacer comparaciones desde el año 2012 en la mayoría de los casos. En la estación M0003 (Figura 6D) ubicada la provincia de Pichincha, el IMERG V06 está sobreestimado, la lluvia observada cuenta con un pico notorio, el cual la precipitación satelital no detecta. La estación M0006 (Figura 6E) ubicada la provincia de Los Ríos, es la estación más parecida solamente se presencia un desfase horizontal en la medición, sin embargo claramente se puede notar la similitud. La estación M0007 (Figura 6F) ubicada la provincia de Orellana, es la serie que tiene menos cantidad de datos, dificultando el hallar alguna relación. En la estación M0025 (Figura 6G) ubicada la provincia de Esmeraldas, el IMERG V06 está sobreestimado excepto en un pico que llega hasta los 800 mm debido a que en esta estación existía mediciones erróneas, el control de calidad hizo su trabajo pero aun así resultó una lluvia extrema. La última estación es la M0033 (Figura 6H) ubicada la provincia de Loja, en la cual el IMERG V06 no hay gran diferencia, pero IMERG V06 está sobreestimado excepto por un pico.

3.1.2 Validación IMERG V06

A continuación, se presentaran 4 gráficas una por cada una de las métricas estadísticas de validación. El Coeficiente de correlación de Pearson y NSE son adimensionales, sin embargo PBIAS su magnitud es el porcentaje y RMSE tiene unidades de los valores con los que se está trabajando, en este caso mm. En el eje de las abscisas está el nombre de las estaciones, en el eje de las ordenadas se dispondrá las métricas y el color de la barra representa la duración de la serie a la que pertenece el dato.

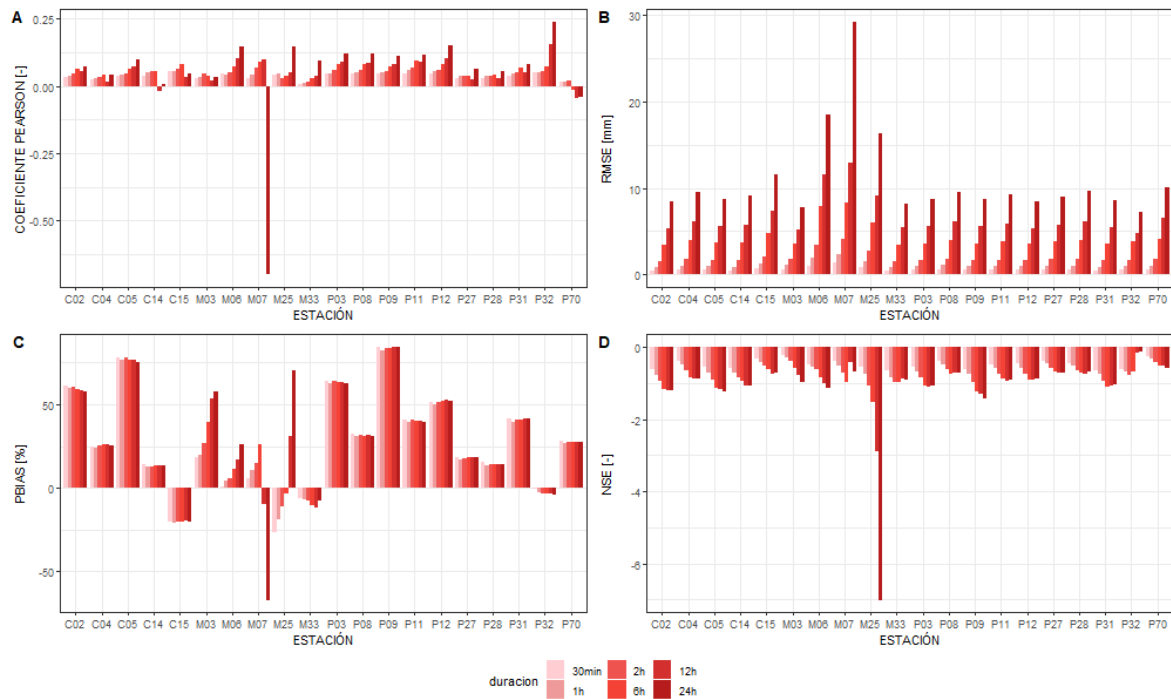


Figura 7. Validación con Coeficiente de correlación de Pearson, RMSE, PBIAS y NSE de los datos IMERG V06 vs Observados para distintas duraciones.

Elaboración: propia

Coeficiente de correlación de Pearson

El valor perfecto para el coeficiente de Pearson es 1, el valor más cercano es 0.24 en la (Figura 7A) de la estación P32 para su máxima duración (24 horas). El peor valor es -0.7 de la estación M0007, esto puede deberse a la casi nula cantidad de datos. Los resultados con valores negativos se los tiene que tomar como resultados insatisfactorios. Las estaciones que tienen resultados insatisfactorios son C14, M0007 y P70. Existe una tendencia en la que el valor de correlación va aumentando conforme aumenta el valor de la duración, exceptuando algunos casos en la duración de 12 horas en las estaciones C02, C04, C14, P27, P28, P31 y M0003.

Según Rodríguez-Sandoval et al., (2023) para la precipitación mensual y anual se obtuvo mejores resultados que para la diaria entonces el Coeficiente de Correlación de Pearson es directamente proporcional a la duración, lo que en la mayoría de nuestras estaciones se cumple, exceptuando la estación P70. Esto puede atribuirse a que cuando la topografía y el clima es complejo no se registran adecuadamente las precipitaciones. En Lau & Behrangi, (2022) existe igualmente una relación directamente proporcional entre el coeficiente de correlación de Pearson y la duración, dando mejores resultados para duraciones mayores (24 horas), sin embargo hay casos en los que mientras las duraciones aumentan el coeficiente de correlación de Pearson disminuye, que resulta similar a las excepciones de este caso.

RMSE

La raíz del error medio cuadrático es adecuada cuando más se acerca al valor de cero. En la (Figura 7B) es fácil notar un aumento del RMSE directamente proporcional a la duración, dando como resultado mayores errores en la duración de 24 horas en todas las estaciones. En las estaciones del INAMHI se presenta la mayor cantidad de errores comparado con en las estaciones del EPMAPS, esto puede deberse a la poca cantidad de datos que se tenía, por ejemplo la estación M0007 que es la de menos datos y la de mayor error 29,2. La estación que mejor resultados arrojó fue la P32 en todas sus duraciones.

En Kyaw et al., (2022), el promedio de RMSE para IMERG fue de 18.44 mostrando valores altos en comparación con otros satélites, en nuestro caso también se obtuvo valores altos de RMSE para las duraciones más grandes. Por otra parte en Rodríguez-Sandoval et al., (2023), IMERG V06 brindó mejores resultados que otros productos satelitales, dando valores mínimos de error, resultados contrarios a los nuestros.

PBIAS

En la (Figura 7C) PBIAS los mejores resultados se dan mientras el porcentaje este más próximo a cero, para este estudio se obtuvo un rango de resultados de -85 a 80%. Si el porcentaje es positivo existe una sobreestimación y por el contrario cuando el porcentaje es negativo existe una subestimación de los datos simulados (IMERG V06) con respecto a las observaciones. Por lo que las mejores estaciones son P32 (subestimada-muy bueno), M0033 (subestimada-bueno), C14 (sobrestimada-bueno) y las estaciones con peores resultados son P09 (sobrestimada-insatisfactorio), C05 (sobrestimada-insatisfactorio) y M0007 (subestimada-insatisfactorio).

La mayoría de los resultados del PBIAS en nuestro estudio se encuentran dentro del rango del 0 al 30%. En Kyaw et al., (2022), el promedio de IMERG es -66.9% un resultado insatisfactorio, dándose frecuentemente una subestimación de la precipitación simulada caso opuesto a nuestro estudio ya que en la mayoría de estaciones existe una sobreestimación de las precipitaciones del IMERG V06. Para Lau & Behrangi, (2022) en la zona de Estados Unidos se presenta valores cercanos a cero mientras la duración aumenta, pero en el mismo estudio en la zona de Austria las duraciones medias (2 a 3 horas) tienen los resultados más cercanos a cero. Resultando valores más similares de nuestro estudio con los valores de PBIAS para la zona de Austria.

NSE

La varianza de los datos simulados respecto a la varianza residual de los datos observados tiene una relación directamente proporcional del resultado de este coeficiente a la duración. En la (Figura 7D) la mayoría de los resultados de NSE se encuentra en el rango entre [-0.5 ; -0.1], la estación M0025 tiene un valor raro de -7 que se encuentra fuera del rango aceptable lo que lo convertiría en un resultado despreciable. La estación que tiene resultados satisfactorios es la M0003 que tiene el mejor resultado para la duración de 24 horas $NSE = 0.99$.

En las cuatro métricas de validación una estación es la que sobresale por las demás obteniendo en casi todas las pruebas los mejores resultados y es la P32 la que más se acerca a uno en el Coeficiente de Pearson, una de las que más se acerca a cero en RMSE, tiene calificación de muy bueno en PBIAS y en NSE tiene resultados buenos. Sin embargo, la estación con peores resultados es la M0007 que cuenta con resultados insatisfactorios en las tres primeras métricas, esto sin duda es debido a la poca cantidad de datos y al gran porcentaje de vacíos.

3.1.3 Series máximas anuales

Las series máximas anuales (AMS) se representaron en gráficos de barras, debido a que las series tienen diferentes duraciones serán diferentes ya que la agregación cambia los valores. Se dividirá nuevamente las imágenes por cada píxel del raster. En el eje de las abscisas se ubicará los años del registro de datos (2000-2021), en el eje de las ordenadas está el mayor valor de precipitación medidos en mm en ese año para cada duración. Las barras tendrán diferente color, cuando sea azul representara a los datos satelitales de IMERG V06 y cuando sea rojo o cualquier color de la paleta de colores roja representara una los datos observados ya sea INAMHI o EPMAPS.

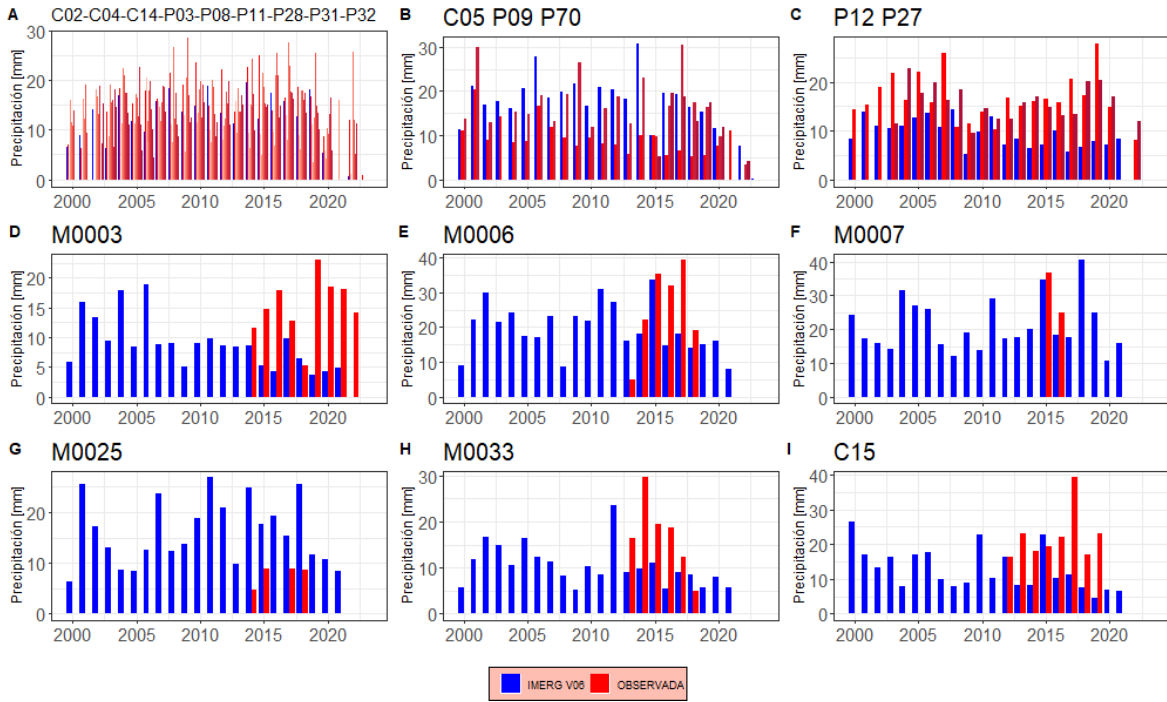


Figura 8. Serie Máxima Anual con datos de 30 minutos IMERG V06 vs Observadas.

Elaboración: propia

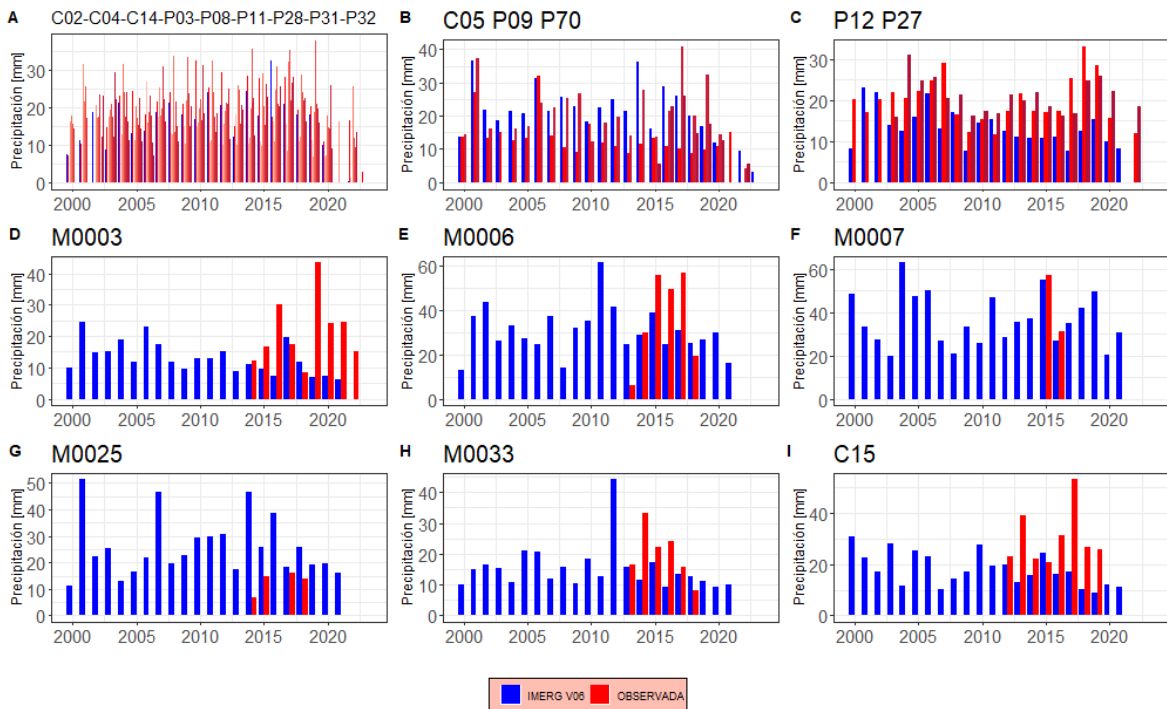


Figura 9. Serie Máxima Anual con datos de 1 hora IMERG V06 vs Observadas.

Elaboración: propia

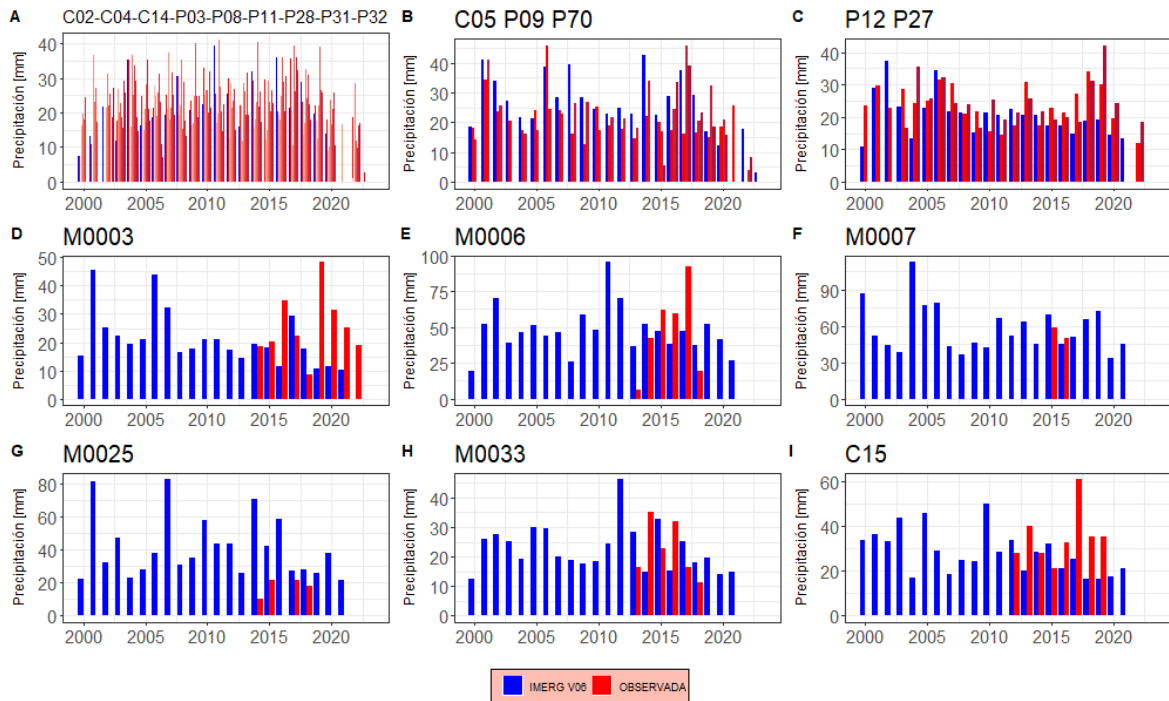


Figura 10. Serie Máxima Anual con datos de 2 horas IMERG V06 vs Observadas.

Elaboración: propia

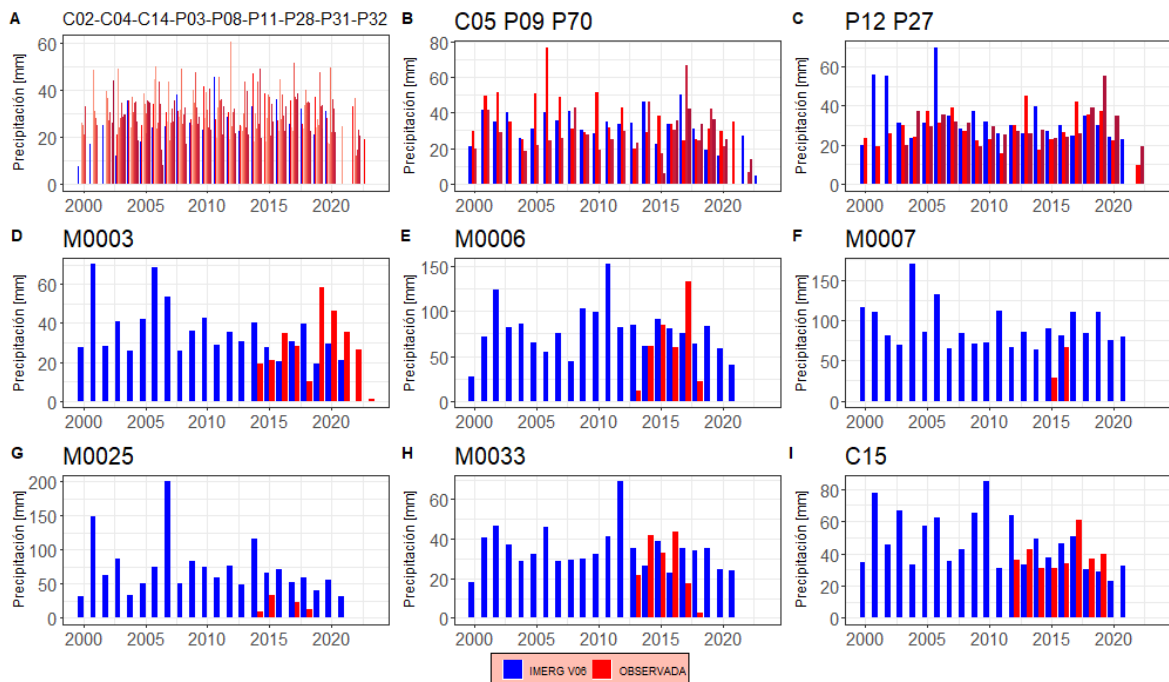


Figura 11. Serie Máxima Anual con datos de 6 horas IMERG V06 vs Observadas.

Elaboración: propia

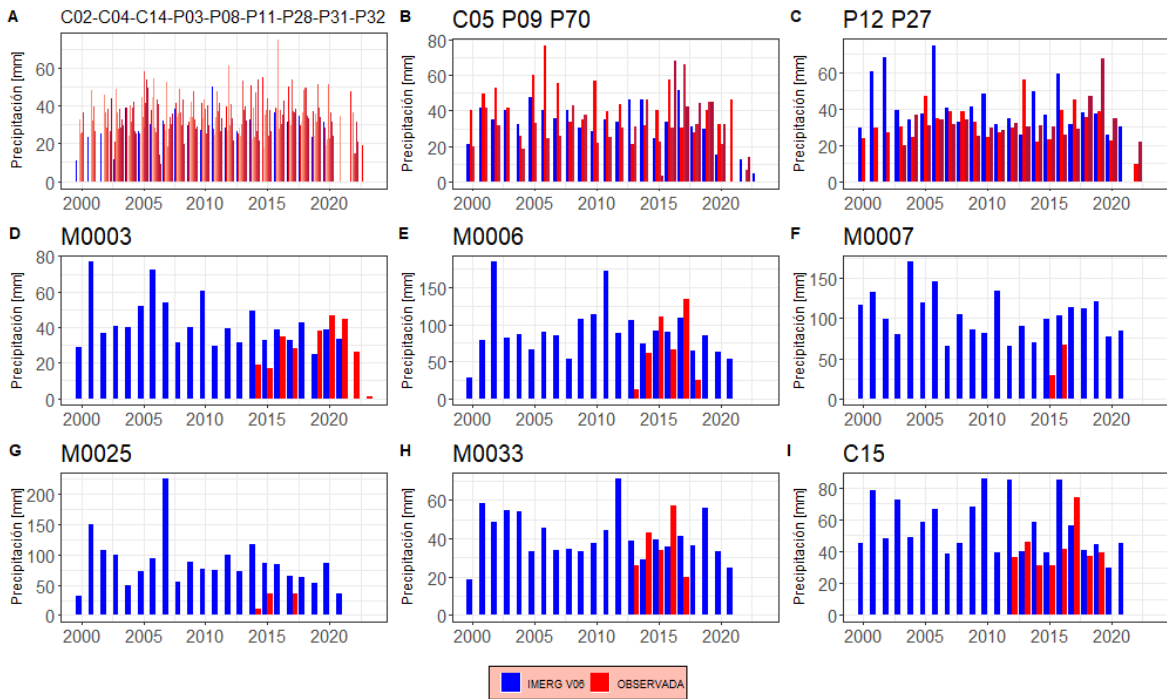


Figura 12. Serie Máxima Anual con datos de 12 horas IMERG V06 vs Observadas.

Elaboración: propia

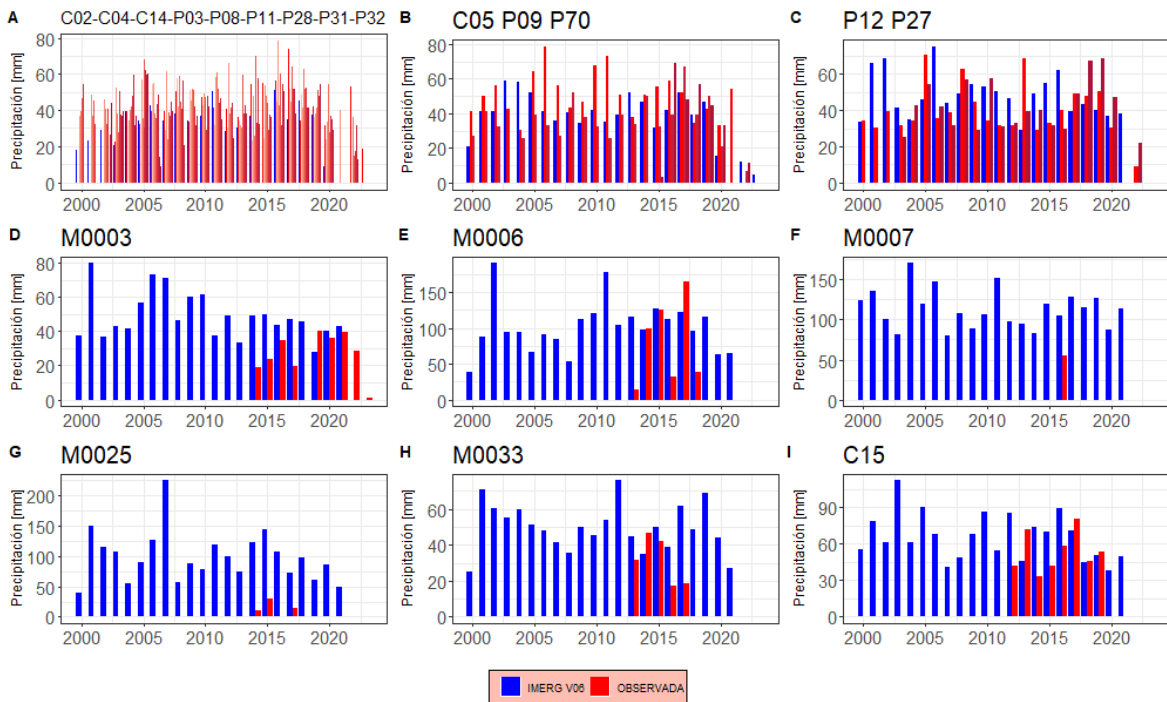


Figura 13. Serie Máxima Anual con datos de 24 horas IMERG V06 vs Observadas.

Elaboración: propia

Existe una relación directa notoria entre el incremento de la duración con la similitud del IMERG V06 con los datos observados, sin embargo, mientras aumenta la duración algunos datos pueden llegar a perderse. La estación M0003 ubicada en la Sierra centro de Ecuador a 3058 m s. n. m. en su graficas de series anuales (Figuras 8D, 9D, 10D, 11D, 12D, 13D) disminuye la subestimación de la precipitación satelital en cuanto se aumenta la duración de la serie, entregando mejores resultados para la serie de 24 horas. El coeficiente de Pearson, el RMSE, el PBIAS tiene resultados insatisfactorios para esta estación, los cuales se agravan aún más para las duraciones más grandes, lo que resulta contradictorio con las series máximas anuales. Por el contrario, el NSE arroja resultados buenos para esta estación.

La estación M0006 (Figuras 8E, 9E, 10E, 11E, 12E, 13E) ubicada en la Costa centro de Ecuador a 120 m s. n. m. disminuye la subestimación de la precipitación satelital en cuanto aumenta la duración de la serie entregando mejores resultados para la serie de 12 horas de duración. El coeficiente de Pearson tiene resultados insatisfactorios para esta estación excepto para la duración de 12 y 24 horas, lo que resulta similar con las AMS. Por el contrario, PBIAS y RMSE dan valores inadecuados que se agravan para las duraciones más grandes, lo que contradice los resultados. NSE arroja resultados buenos para 12 horas para las demás duraciones tiene mucho error.

La estación M0007 (Figuras 8F, 9F, 10F, 11F, 12F, 13F) ubicada en la parte límite noroccidental de Ecuador a 185 m s. n. m. disminuye la subestimación de la precipitación satelital, además se ve una disminución en la cantidad de datos en cuanto aumente la duración de la serie. La estación entrega mejores resultados para la serie de 2 horas de duración. Todos las métricas de validación dan malos resultados principalmente las series de 12 y 24 horas. No existe confianza en sus valores debido a la escasez de datos.

La estación M0025 (Figuras 8G, 9G, 10G, 11G, 12G, 13G) ubicada en la Costa centro de Ecuador a 379 m s. n. m. aumenta la sobreestimación de la precipitación satelital en cuanto aumente la duración de la serie, entregando mejores resultados para la serie de 2 horas de duración. Para duraciones de 12 y 24 horas hay una mala correlación con las variables de validación, dando resultados malos y esto se asemeja al AMS ya que no tiene ningún parecido comparándolas con el AMS de IMERG V06. El coeficiente de Pearson, PBIAS, RMSE y NSE tiene resultados aceptables para la duración de 2 horas lo que se asemeja con el grafico de la serie máxima de dos horas que resulta la más similar.

La estación M0033 (Figuras 8H, 9H, 10H, 11H, 12H, 13H) ubicada en la Costa centro de Ecuador a 2160 m s. n. m. nuevamente disminuye la subestimación de la precipitación

satelital en cuanto aumente la duración de la serie entregando mejores resultados para la serie de 24 horas de duración. El coeficiente de Pearson tiene resultados insatisfactorios para esta estación. Por el contrario RMSE da buenos resultados excepto para la duración de 24 horas, PBIAS y NSE da valores adecuados para todas las duraciones.

A continuación, se realizará el análisis de las estaciones de EPMAPS, la estación C15 (Figuras 8I, 9I, 10I, 11I, 12I, 13I) se encuentra localizada sola dentro de un píxel del DMQ a 1767 m s. n. m. sus valores son parecidos para todas la agregaciones dando la mayor similitud para la duración de 6 horas. Es una de las pocas estaciones, en las que mientras crece la duración disminuye su coeficiente de Pearson dando malos resultados para las duraciones altas. Por otra parte RMSE y NSE tiene los peores resultados para duraciones altas, en cambio PBIAS arroja resultados similares y buenos en cualquier duración.

Para P12-P27 (Figuras 8C, 9C, 10C, 11C, 12C, 13C) existe una disminución en la subestimación de la precipitación satelital mientras crece la duración, dando para ambas estaciones los mejores resultados de AMS para la duración de 24 horas. La estación P12 está localizada 3232 m s. n. m. y P27 está localizada a 3551 m s. n. m., en el Coeficiente de Pearson mientras aumenta la duración mejores son los resultados excepto para la estación P27 en 12 horas. En RMSE las estaciones trabajan de manera parecida, pero dan malos resultados para las duraciones altas. En PBIAS la estación P27 tiene un mejor comportamiento a comparación de P12, por otro lado P12 arrojo resultados satisfactorios para la serie máxima anual siendo muy parecida a la serie observada, mejores que los resultados de P27.

En las (Figuras 8B, 9B, 10B, 11B, 12B, 13B) de las estaciones C05-P09-P70 de EPMAPS tienen los mejores y similares resultados de AMS para la duración de 2 horas. La estación C05 está localizada 2960 m s. n. m., la estación P09 está localizada 2804 m s. n. m. y P70 está localizada a 2903 m s. n. m., para el Coeficiente de Pearson mientras aumenta la duración mejores son los resultados, excepto para la estación P70 que cuenta con valores negativos que hacen despreciables los resultados de esta estación. En RMSE las estaciones trabajan de manera parecida pero los mejores resultados aparecen en las menores duraciones. En PBIAS todas las duraciones tienen un comportamiento similar, la estación que tiene mejores resultados es P70, luego C05, y por ultimo P09. En NSE P70 tiene resultados buenos, P09 y C05 tienen resultados satisfactorios para duraciones hasta 6 horas, por el contrario 12 y 24 horas los resultados sobrepasan el rango normal.

En las (Figuras 8A, 9A, 10A, 11A, 12A, 13A) de las estaciones restantes de EPMAPS (C02, C04, C14, P03, P08, P11, P28, P31, P32) que se encuentran en el mismo píxel, mientras

aumenta las duraciones aumenta la subestimación de precipitación satelital. La estación P32 resulto ser la mejor en el anterior análisis, por ello se realizará énfasis en esta. El Coeficiente de Pearson, RMSE al igual que la AMS para P32 dieron los mejores resultados para duración de 24 horas. En PBIAS sin duda es la estación con menos error en todas sus duraciones, y en NSE para duraciones de 12 y 24 horas tienen resultados insatisfactorios.

3.1.4 Análisis del ajuste funciones de distribución de probabilidad

Las 40 estaciones (20 observadas y 20 satelitales) fueron ajustadas en todas sus duraciones a las funciones de probabilidad de: Gumbel, Log Normal, Pearson III y Log Pearson III. El siguiente paso es el análisis para seleccionar cuál de estas funciones se ajusta mejor realizando las pruebas de Chi-Cuadrado y Kolmogórov-Smirnov, a esta última se le dará más peso a la hora de seleccionar la función. Obteniendo el siguiente cuadro resumen:

Tabla 14. Función de distribución que mejor se ajusta por la prueba Kolmogórov-Smirnov.

#	NOMBRE	ORIGEN	DURACIONES					
			30 m	1 h	2 h	6 h	12 h	24 h
M0003	Izobamba	INAMHI	Pearson III	Pearson III	Pearson III	Pearson III	Pearson III	Pearson III
M0006	Pichilingue	INAMHI	Pearson III	Log Pearson III	Pearson III	Pearson III	Pearson III	Log Pearson III
M0007	Nuevo Rocafuerte	INAMHI						
M0025	La Concordia	INAMHI	Pearson III	Pearson III	Pearson III	Pearson III	Pearson III	Log Pearson III
M0033	La Argelia-Loja	INAMHI	Pearson III	Pearson III	Log Pearson III	Pearson III	Pearson III	Pearson III
C02	Rumihurco	EPMAPS	Pearson III	Pearson III	Gumbel	Pearson III	Pearson III	Pearson III
C04	Rumipamba	EPMAPS	Pearson III	Pearson III	Pearson III	Pearson III	Pearson III	Pearson III
C05	Bellavista	EPMAPS	Pearson III	Pearson III	Pearson III	Pearson III	Pearson III	Pearson III
C14	Mindo Captación	EPMAPS	Pearson III	Pearson III	Pearson III	Pearson III	Pearson III	Pearson III
C15	Nanegalito	EPMAPS	Pearson III	Pearson III	Pearson III	Pearson III	Log Pearson III	Pearson III
P03	Rumihurco Machángara	EPMAPS	Pearson III	Pearson III	Log Pearson III	Pearson III	Log Pearson III	Log Normal
P08	Rumipamba Bodegas	EPMAPS	Pearson III	Pearson III	Log Pearson III	GUMBEL	Log Pearson III	Pearson III
P09	Iñaquito INAMHI	EPMAPS	Pearson III	Pearson III	Log Normal	Pearson III	Pearson III	Pearson III

P11	Antenas	EPMAPS	Pearson III	Log Pearson III	Gumbel	Pearson III	Pearson III	Pearson III
P12	Toctiuco	EPMAPS	Pearson III	Pearson III	Log Pearson III	Pearson III	Pearson III	Log Normal
P27	San Francisco	EPMAPS	Pearson III	Pearson III	Pearson III	Pearson III	GUMBE L	Pearson III
P28	Cruz Loma	EPMAPS	Pearson III	Pearson III	Pearson III	Pearson III	Pearson III	Pearson III
P31	Pichán	EPMAPS	Pearson III	Pearson III	Pearson III	Pearson III	Pearson III	Pearson III
P32	Mindo Bajo	EPMAPS	Pearson III	Log Pearson III	Pearson III	Pearson III	Pearson III	Pearson III
P70	CC El Bosque	EPMAPS	Pearson III	Pearson III	Pearson III	Pearson III	Pearson III	Pearson III
M0003	Izobamba	IMERG V06	Pearson III	Pearson III	Log Pearson III	Pearson III	Pearson III	Pearson III
M0006	Pichilingue	IMERG V06	Pearson III	Pearson III	Pearson III	Pearson III	Pearson III	Pearson III
M0007	Nuevo Rocafuerte	IMERG V06	Pearson III	Pearson III	Pearson III	Pearson III	Pearson III	Pearson III
M0025	La Concordia	IMERG V06	Pearson III	Pearson III	Pearson III	Log Pearson III	Log Pearson III	Pearson III
M0033	La Argelia- Loja	IMERG V06	Pearson III	Log Pearson III	Pearson III	Log Pearson III	Pearson III	Pearson III
C02	Rumihurco	IMERG V06	Log Pearson III	Log Pearson III	Pearson III	Pearson III	Pearson III	Pearson III
C04	Rumipamba	IMERG V06	Log Pearson III	Log Pearson III	Pearson III	Pearson III	Pearson III	Pearson III
C05	Bellavista	IMERG V06	Log Pearson III	Log Pearson III	Log Pearson III	Pearson III	Pearson III	Pearson III
C14	Mindo Captación	IMERG V06	Log Pearson III	Log Pearson III	Log Pearson III	Pearson III	Pearson III	Pearson III
C15	Nanegalito	IMERG V06	Log Pearson III	Pearson III	Log Pearson III	Log Pearson III	Log Pearson III	Pearson III
P03	Rumihurco Machángara	IMERG V06	Log Pearson III	Log Pearson III	Log Pearson III	Pearson III	Log Pearson III	Log Pearson III
P08	Rumipamba Bodegas	IMERG V06	Log Pearson III	Log Pearson III	Log Pearson III	Pearson III	Log Pearson III	Log Pearson III
P09	Iñaquito INAMHI	IMERG V06	Log Pearson III	Log Pearson III	Log Pearson III	Pearson III	Log Pearson III	Log Pearson III
P11	Antenas	IMERG V06	Log Pearson III	Log Pearson III	Log Pearson III	Pearson III	Log Pearson III	Log Pearson III
P12	Toctiuco	IMERG V06	Pearson III	Log Pearson III	Log Pearson III	Pearson III	Log Pearson III	Log Pearson III
P27	San Francisco	IMERG V06	Pearson III	Log Pearson III	Log Pearson III	Pearson III	Pearson III	Log Pearson III

P28	Cruz Loma	IMERG V06	Log Pearson III	Log Pearson III	Log Pearson III	Pearson III	Log Pearson III	Log Pearson III
P31	Pichán	IMERG V06	Log Pearson III	Log Pearson III	Log Pearson III	Pearson III	Log Pearson III	Log Pearson III
P32	Mindo Bajo	IMERG V06	Log Pearson III	Log Pearson III	Log Pearson III	Pearson III	Log Pearson III	Log Pearson III
P70	CC El Bosque	IMERG V06	Log Pearson III	Log Pearson III	Log Pearson III	Log Pearson III	Log Pearson III	Log Pearson III

Elaboración: propia

Los parámetros de las funciones se obtuvieron mediante el método de los momentos, resultando en que la mayoría de las estaciones se ajustan a Pearson III, seguida de Log Pearson III y en menor cantidad Gumbel y Log Normal en ese orden. En Kyaw et al., (2022) GEV mostró mejores resultados sobre las demás funciones como Gumbel que tenía valores muy similares a GEV y Log Pearson III fue la función con los peores resultados. Teniendo resultados opuestos a este estudio, donde Log Pearson III tuvo mejores resultados. Para Lau & Behrangi, (2022) IMERG podía ajustarse adecuadamente a cualquiera de las cuatro CDF (valor $p > 0,05$). Sin embargo, la prueba de Kolmogórov-Smirnov dice que GEV con MLE proporcionan la mejor bondad de ajuste (valor p máximo = 0,89) para todos los conjuntos de datos en ambas regiones. En segundo lugar se encuentra la distribución de Gumbel con MLE (valor p máximo = 0,87), y Pearson 3 obtuvo la peor bondad de ajuste (valor p máximo = 0,81) para todos los conjuntos de datos en ambas regiones. Los resultados coinciden con Kyaw et al., (2022) pero no con este estudio, esto puede estar relacionado con el hecho de que la selección de la distribución adecuada está influida por la longitud del registro, además de la área de estudio que es muy diferente a la nuestra por diversos factores como la topografía, el clima, calidad de estaciones, etc.

En Colombia, con Rodríguez-Sandoval et al., (2023) el análisis se dio ajustando las series con la función de distribución de Gumbel debido a que el Instituto de Hidrología Meteorología y Estudios Ambientales de Colombia (IDEAM) ocupa esta misma función para realizar sus curvas IDF, obviamente para que sea más parecido los resultados se ocupó esta función, dando mejores resultados en la comparación.

3.1.5 Curvas IDF

Una vez definida la función de distribución de probabilidad con mejor ajuste, las duraciones y el periodo de retorno seleccionado 2 a 50 años podemos obtener los resultados de precipitación porque hemos estado trabajando con estos datos. En las estaciones INAMHI ocuparemos los valores ajustados proporcionados por esta institución de las intensidades

observadas para diferentes duraciones y periodos de retorno. Para obtener la intensidad de las curvas IDF, la precipitación se la debe dividir para su duración, las curvas se han repartido para su comparación con la anterior división por pixeles. En el eje de las abscisas se dispondrá las duraciones, en el eje de las ordenadas se ubicará las intensidades, por el color de línea se distinguirá si es observada o satelital y el tipo de línea definirá su periodo de retorno, así:

Tabla 15. Intensidades en mm/h para periodos de retorno (Frecuencias) para cada duración, datos de curvas IDF para la estación M0003 IMERG V06 vs Observada.

		DURACIÓN						
FRECUENCIA	IDF	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr	ORIGEN
	2	16.28	12.37	9.32	5.43	3.27	1.95	IMERG V06
	5	24.13	17.07	13.39	7.56	4.33	2.46	IMERG V06
	10	29.16	19.94	16.51	9.01	5.04	2.77	IMERG V06
	50	39.63	25.68	24.67	12.12	6.56	3.41	IMERG V06
	2	39.18	25.55	14.73	5.71	3.02	1.67	INAMHI
	5	48.52	31.91	18.78	7.04	3.75	2.1	INAMHI
	10	54.26	35.84	21.14	7.86	4.21	2.39	INAMHI
	50	66.08	43.95	26.75	9.53	5.19	3.02	INAMHI

INTENSIDAD

Fuente: INAMHI

Elaboración: propia

M0003

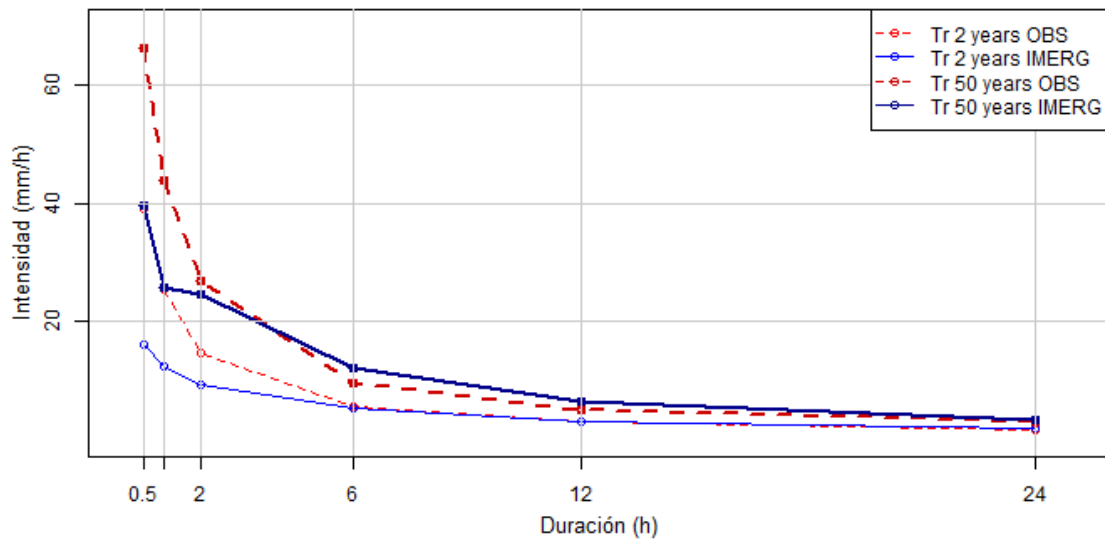


Figura 14. Curvas IDF para la estación M0003 IMERG V06 vs Observada.

Elaboración: propia

Tabla 16. Intensidades en mm/h para periodos de retorno (Frecuencias) para cada duración, datos de curvas IDF para la estación M0006 IMERG V06 vs Observada.

		DURACIÓN						
FRECUENCIA	IDF	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr	ORIGEN
	2	38.52	29.15	22.45	12.35	6.86	4.04	IMERG V06
	5	50.69	38.94	30.05	16.54	9.59	5.40	IMERG V06
	10	57.38	44.98	34.88	19.08	11.41	6.22	IMERG V06
	50	69.66	57.22	44.86	24.15	15.27	7.87	IMERG V06
	2	72.33	54.03	35.54	16.24	8.57	4.73	INAMHI
	5	89.47	67.61	45.18	20.29	10.7	5.91	INAMHI
	10	100	76.02	51.22	22.8	12.03	6.64	INAMHI
	50	121.55	93.38	63.84	27.96	14.75	8.15	INAMHI

INTENSIDAD

Fuente: INAMHI

Elaboración: propia

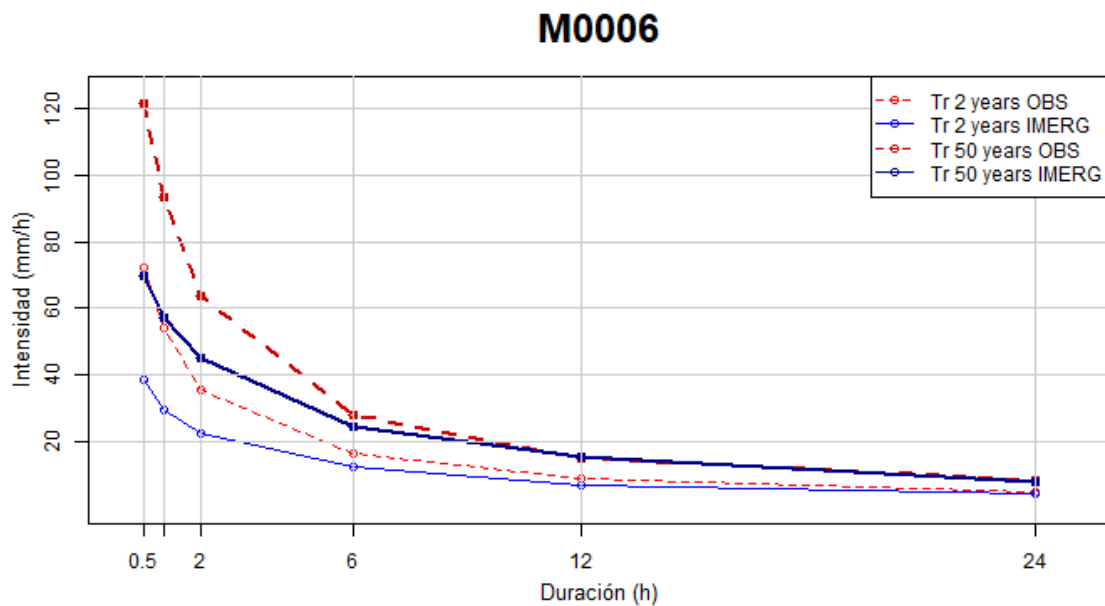


Figura 15. Curvas IDF para la estación M0006 IMERG V06 vs Observada.

Elaboración: propia

Tabla 17. Intensidades en mm/h para periodos de retorno (Frecuencias) para cada duración, datos de curvas IDF para la estación M0007 IMERG V06 vs Observada.

		DURACIÓN						
FRECUENCIA	IDF	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr	ORIGEN
	2	39.60	35.56	26.97	14.24	8.37	4.59	IMERG V06
	5	53.84	46.36	36.04	18.23	10.41	5.50	IMERG V06
	10	62.82	52.56	41.95	20.98	11.63	6.04	IMERG V06
	50	81.28	64.39	54.40	27.01	14.01	7.11	IMERG V06
	2	69.28	46.38	29.05	12.11	6.41	3.56	INAMHI
	5	84.03	57.18	35.43	14.88	7.85	4.33	INAMHI
	10	92.96	63.79	39.3	16.57	8.72	4.79	INAMHI
	50	110.98	77.3	47.15	20.01	10.48	5.72	INAMHI

INTENSIDAD

Fuente: INAMHI

Elaboración: propia

M0007

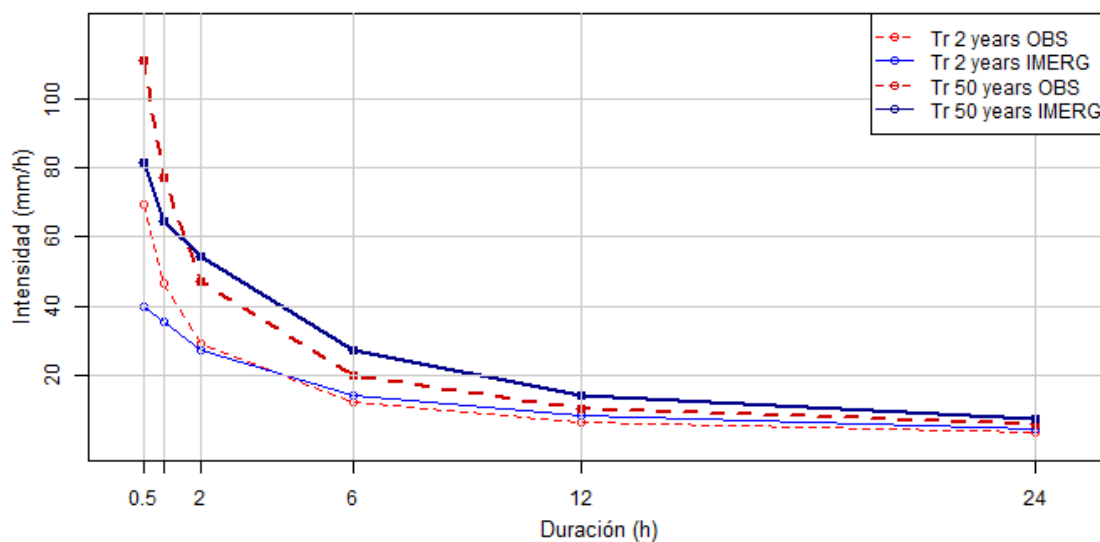


Figura 16. Curvas IDF para la estación M0007 IMERG V06 vs Observada.

Elaboración: propia

Tabla 18. Intensidades en mm/h para periodos de retorno (Frecuencias) para cada duración, datos de curvas IDF para la estación M0025 IMERG V06 vs Observada.

		DURACIÓN						
FRECUENCIA	IDF	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr	ORIGEN
	2	31.33	23.92	18.80	10.19	6.44	3.74	IMERG V06
	5	42.61	34.14	27.33	15.45	9.33	5.34	IMERG V06
	10	48.91	40.69	32.86	19.80	11.41	6.40	IMERG V06
	50	60.65	54.32	44.48	32.24	16.42	8.67	IMERG V06
	2	78.32	61.17	40.18	19.64	10.31	5.64	INAMHI
	5	107.72	82.53	44.54	24.49	12.9	7.11	INAMHI
	10	126.92	92.52	63.15	26.99	14.38	8.08	INAMHI
50	169.17	127.05	82.92	31.33	17.18	10.11	INAMHI	

INTENSIDAD

Fuente: INAMHI

Elaboración: propia

M0025

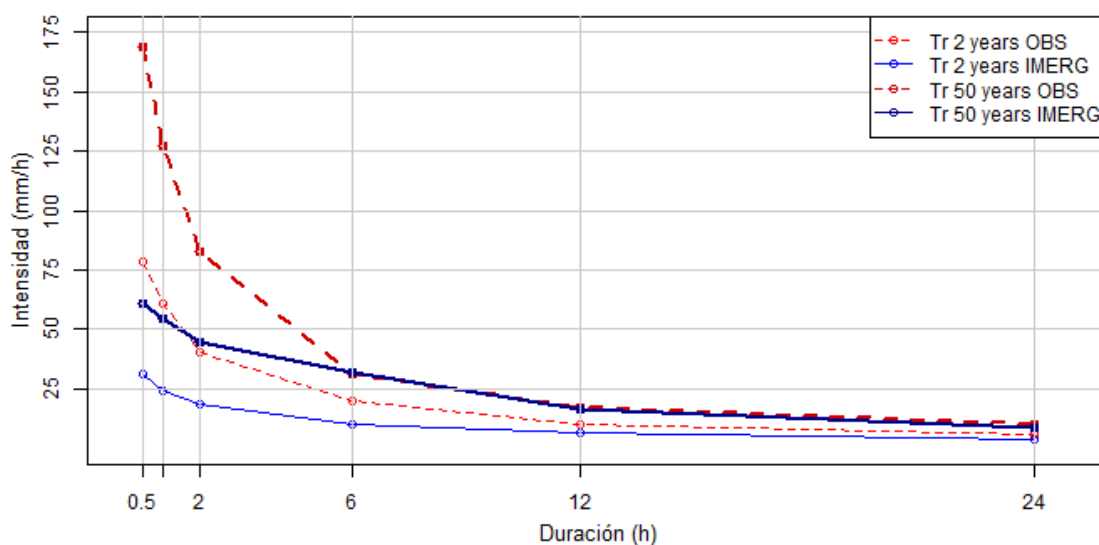


Figura 17. Curvas IDF para la estación M0025 IMERG V06 vs Observada.

Elaboración: propia

Tabla 19. Intensidades en mm/h para periodos de retorno (Frecuencias) para cada duración, datos de curvas IDF para la estación M0033 IMERG V06 vs Observada.

		DURACIÓN						
IA UENC FREC	IDF	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr	ORIGEN
	2	18.77	12.92	10.56	5.43	3.32	2.06	IMERG V06

5	27.04	17.98	14.23	6.96	4.23	2.53	IMERG V06
10	32.61	22.65	16.65	8.01	4.77	2.78	IMERG V06
50	44.64	37.81	21.79	10.42	5.82	3.24	IMERG V06
2	32.76	21.66	12.73	5.83	3.13	1.78	INAMHI
5	42.87	28.18	16.44	7.01	3.85	2.27	INAMHI
10	49.56	32.49	18.9	7.72	4.31	2.6	INAMHI
50	64.29	41.99	24.31	9.15	5.26	3.31	INAMHI

INTENSIDAD

Fuente: INAMHI

Elaboración: propia

M0033

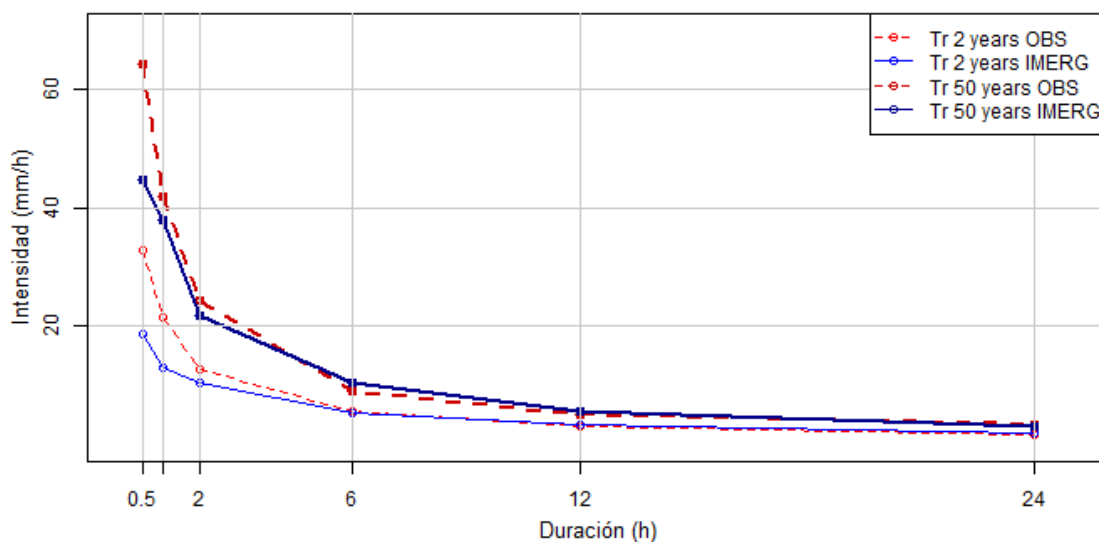


Figura 18. Curvas IDF para la estación M033 IMERG V06 vs Observada.

Elaboración: propia

Tabla 20. Intensidades en mm/h para periodos de retorno (Frecuencias) para cada duración, datos de curvas IDF para la estación C15 IMERG V06 vs Observada.

		DURACIÓN						
FRECUENCIA	IDF	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr	ORIGEN
	2	22.53	17.56	13.12	7.27	4.37	2.64	IMERG V06
	5	33.75	23.29	17.57	9.90	5.67	3.36	IMERG V06
	10	41.96	26.54	20.61	11.69	6.54	3.78	IMERG V06
	50	62.10	32.71	27.55	15.81	8.49	4.60	IMERG V06
	2	39.91	27.45	16.08	6.00	3.15	2.16	EPMAPS C15

5	53.19	37.66	21.59	7.47	4.06	2.77	EPMAPS C15
10	63.49	44.80	25.44	8.58	4.87	3.13	EPMAPS C15
50	87.78	60.56	33.91	11.14	7.31	3.84	EPMAPS C15

INTENSIDAD

Elaboración: propia

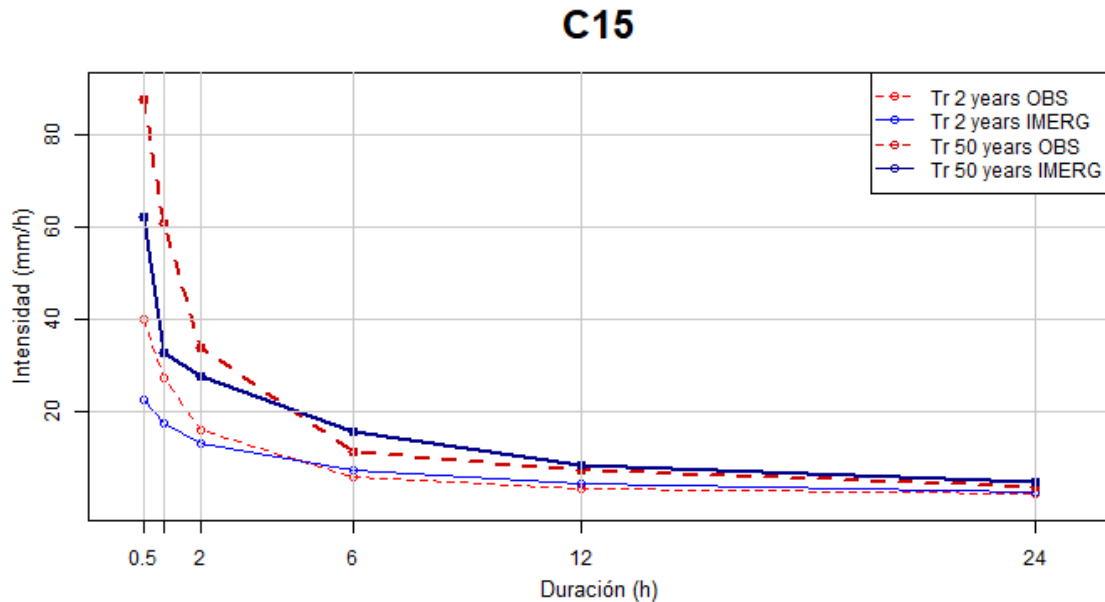


Figura 19. Curvas IDF para la estación C15 IMERG V06 vs Observada.

Elaboración: propia

Tabla 21. Intensidades en mm/h para periodos de retorno (Frecuencias) para cada duración, datos de curvas IDF para la estación P12-P27 IMERG V06 vs Observada.

		DURACIÓN						
FRECUENCIA	IDF	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr	ORIGEN
	2	18.73	12.62	9.64	4.99	3.17	1.93	IMERG V06
	5	23.63	16.70	12.55	6.87	4.15	2.38	IMERG V06
	10	26.36	19.42	14.52	8.25	4.89	2.67	IMERG V06
	50	31.44	25.54	19.00	11.39	6.76	3.28	IMERG V06
	2	32.21	19.19	11.76	4.54	2.60	1.57	EPMAPS P12
	5	40.84	24.31	14.51	5.83	3.35	2.08	EPMAPS P12
	10	45.96	27.36	15.94	6.53	3.77	2.41	EPMAPS P12
	50	56.02	33.36	18.36	7.81	4.54	3.13	EPMAPS P12

2	31.28	20.24	11.50	4.57	2.59	1.70	EPMAPS P27
5	37.48	24.04	14.66	5.90	3.19	2.22	EPMAPS P27
10	40.86	26.35	16.74	6.82	3.58	2.52	EPMAPS P27
50	47.06	30.99	21.16	8.83	4.45	3.11	EPMAPS P27

INTENSIDAD

Elaboración: propia

P12-P27

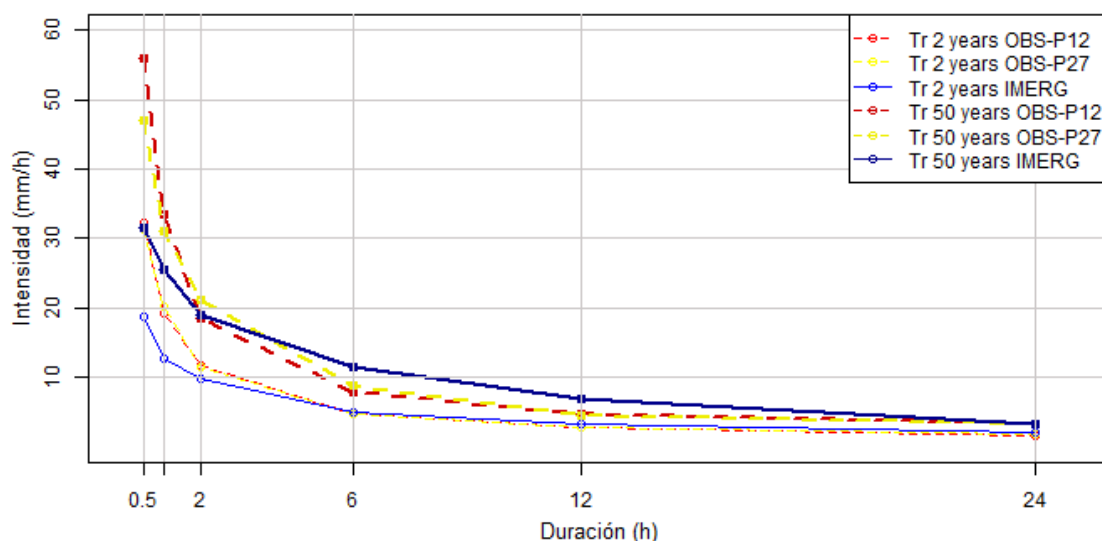


Figura 20. Curvas IDF para la estación P12-P27 IMERG V06 vs Observada.

Elaboración: propia

Tabla 22. Intensidades en mm/h para periodos de retorno (Frecuencias) para cada duración, datos de curvas IDF para la estación C05-P09-P70 IMERG V06 vs Observada.

		DURACIÓN						
FRECUENCIA	IDF	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr	ORIGEN
	2	17.45	11.91	9.54	5.68	3.41	2.07	IMERG V06
	5	23.91	16.16	12.51	7.72	4.43	2.55	IMERG V06
	10	28.64	19.94	14.92	9.11	5.06	2.83	IMERG V06
	50	40.43	31.58	21.59	12.10	6.36	3.39	IMERG V06
	2	35.81	21.38	13.27	5.40	2.85	1.69	EPMAPS C05
	5	45.60	27.95	17.22	6.70	3.57	2.10	EPMAPS C05
	10	50.11	31.34	19.19	7.28	3.88	2.27	EPMAPS C05
	50	57.08	37.22	22.49	8.15	4.34	2.50	EPMAPS C05
	2	32.33	19.95	10.95	4.33	2.44	1.47	EPMAPS P09

5	44.09	27.88	14.97	6.34	3.41	1.93	EPMAPS P09
10	50.89	32.57	17.63	7.68	4.01	2.18	EPMAPS P09
50	63.97	41.77	23.50	10.54	5.22	2.64	EPMAPS P09
2	26.72	14.90	9.91	4.87	3.01	1.65	EPMAPS P70
5	36.42	21.67	15.37	6.48	4.49	2.43	EPMAPS P70
10	40.96	25.28	18.46	7.16	5.24	2.81	EPMAPS P70
50	48.09	31.74	24.28	8.13	6.52	3.41	EPMAPS P70

INTENSIDAD

Elaboración: propia

C05-P09-P70

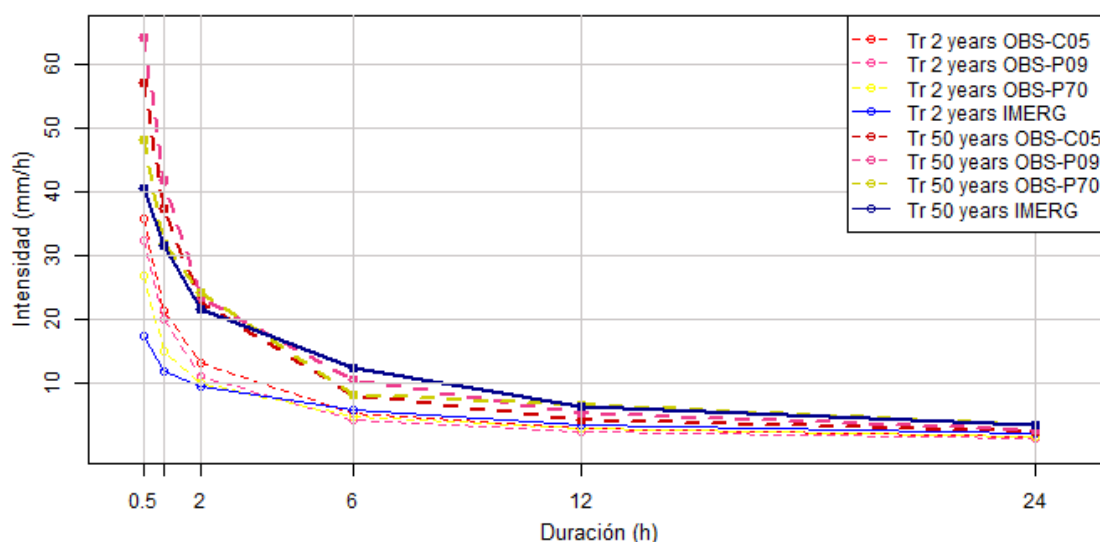


Figura 21. Curvas IDF para la estación C05-P09-P70 IMERG V06 vs Observada.

Elaboración: propia

Tabla 23. Intensidades en mm/h para periodos de retorno (Frecuencias) para cada duración, datos de curvas IDF para la estación C02-C04-C14-P03-P08-P11-P28-P31-P32 IMERG V06 vs Observada.

		DURACIÓN						
FRECUENCIA	IDF	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr	ORIGEN
	2	17.80	12.71	9.06	5.70	3.25	1.95	IMERG V06
	5	26.10	17.80	12.50	7.34	4.22	2.38	IMERG V06
	10	31.64	21.55	14.75	8.29	4.82	2.65	IMERG V06
	50	43.75	30.94	19.51	10.10	6.04	3.19	IMERG V06
	2	26.87	16.96	10.51	4.33	2.44	1.44	EPMAPS C02
	5	34.15	22.56	14.60	5.59	3.04	1.77	EPMAPS C02
	10	37.31	25.34	17.31	6.25	3.35	1.92	EPMAPS C02
	50	41.88	30.01	23.28	7.41	3.88	2.16	EPMAPS C02

2	32.65	20.58	11.96	5.06	2.95	1.71	EPMAPS C04
5	41.57	26.53	15.07	6.08	3.69	2.18	EPMAPS C04
10	45.50	29.30	16.44	6.68	4.09	2.46	EPMAPS C04
50	51.30	33.61	18.45	7.81	4.82	2.98	EPMAPS C04
2	21.53	12.71	8.80	3.66	2.38	1.61	EPMAPS C14
5	29.10	17.67	11.21	4.69	2.91	2.03	EPMAPS C14
10	33.12	20.48	12.27	5.47	3.21	2.26	EPMAPS C14
50	40.28	25.77	13.84	7.31	3.78	2.67	EPMAPS C14
2	34.90	22.66	12.75	5.28	2.92	1.73	EPMAPS P03
5	44.04	27.89	15.63	6.55	3.57	2.11	EPMAPS P03
10	49.14	30.98	17.51	7.29	3.92	2.34	EPMAPS P03
50	58.64	37.02	21.65	8.71	4.54	2.80	EPMAPS P03
2	35.46	22.49	12.81	5.17	3.17	1.95	EPMAPS P08
5	44.43	29.22	16.53	6.87	4.04	2.41	EPMAPS P08
10	49.40	33.15	18.70	7.99	4.47	2.64	EPMAPS P08
50	58.61	40.78	22.82	10.46	5.16	3.02	EPMAPS P08
2	31.46	19.10	11.43	5.09	2.85	1.67	EPMAPS P11
5	36.38	23.04	14.81	5.96	3.35	1.98	EPMAPS P11
10	38.37	25.06	17.04	6.36	3.58	2.13	EPMAPS P11
50	41.01	28.42	21.95	7.01	3.96	2.37	EPMAPS P11
2	30.52	20.93	12.28	5.30	2.93	1.90	EPMAPS P28
5	37.27	25.43	15.03	6.31	3.44	2.23	EPMAPS P28
10	40.52	27.65	16.49	6.88	3.76	2.42	EPMAPS P28
50	45.75	31.34	19.10	7.96	4.39	2.75	EPMAPS P28
2	24.60	16.34	9.79	4.24	2.37	1.45	EPMAPS P31
5	29.78	21.64	12.43	5.19	3.05	1.87	EPMAPS P31
10	32.40	24.86	13.95	5.73	3.47	2.10	EPMAPS P31
50	36.86	31.29	16.86	6.72	4.31	2.50	EPMAPS P31
2	20.87	13.55	9.47	4.58	2.69	1.69	EPMAPS P32
5	29.00	20.78	12.77	6.46	3.76	2.24	EPMAPS P32
10	33.04	25.93	14.37	7.37	4.27	2.48	EPMAPS P32
50	39.77	38.09	16.96	8.88	5.10	2.81	EPMAPS P32

INTENSIDAD

Elaboración: propia

C02-C04-C14-P03-P08-P11-P28-P31-P32

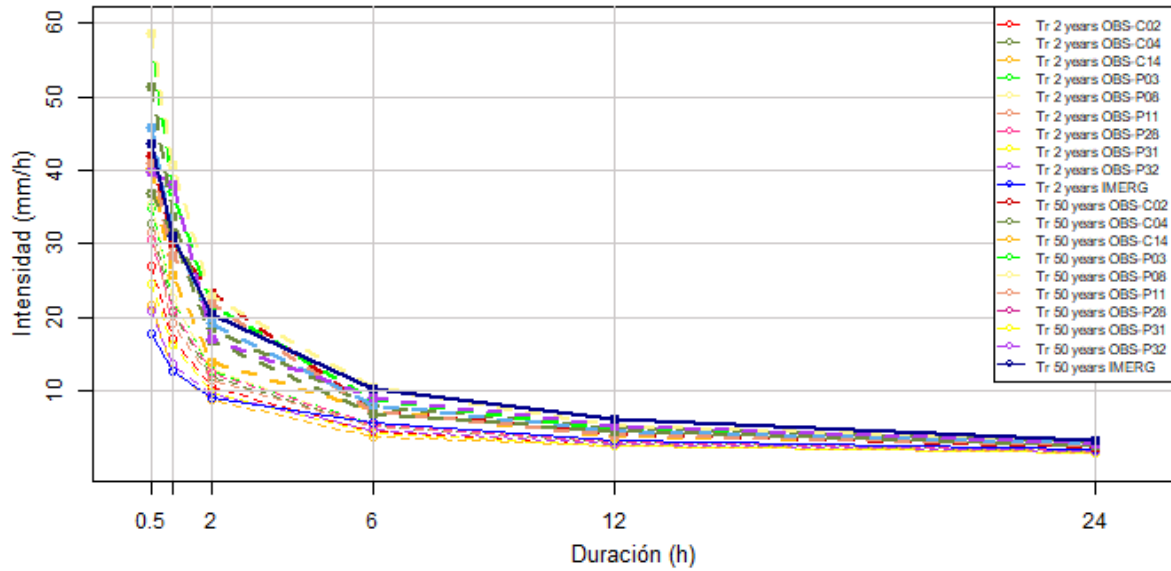


Figura 22. Gráfico N6. Curvas IDF para las estaciones C02-C04-C14-P03-P08-P11-P28-P31-P32 IMERG V06 vs Observada.

Elaboración: propia

Para realizar un análisis minucioso, se obtuvo un PBIAS entre las curvas IDF de IMERG V06 y las observadas, obteniéndose los siguientes resultados de PBIAS y PBIAS promedio para cortas y largas duraciones:

Tabla 24. Error porcentual de las curvas IDF IMERG V06 respecto a las curvas observadas para cada una de las estaciones.

ERROR						
M0003						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-58.45	-51.59	-36.73	-4.90	8.28	16.77
5	-50.27	-46.51	-28.70	7.39	15.47	17.14
10	-46.26	-44.36	-21.90	14.63	19.71	15.90
50	-40.03	-41.57	-7.78	27.18	26.40	12.91
PBIAS (%) =	-48.75	-46.01	-23.78	11.07	17.46	15.68
PBIASprom =		-39.51			14.74	
M0006						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-46.74	-46.05	-36.83	-23.95	-19.95	-14.59
5	-43.34	-42.40	-33.49	-18.48	-10.37	-8.63
10	-42.62	-40.83	-31.90	-16.32	-5.15	-6.33
50	-42.69	-38.72	-29.73	-13.63	3.53	-3.44
PBIAS (%) =	-43.85	-42.00	-32.99	-18.09	-7.99	-8.24

PBIASprom =		-39.61		-11.44		
M0007						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-42.84	-23.33	-7.16	17.59	30.58	28.93
5	-35.93	-18.92	1.72	22.51	32.61	27.02
10	-32.42	-17.60	6.74	26.61	33.37	26.10
50	-26.76	-16.70	15.38	34.98	33.68	24.30
PBIAS (%) =	-34.49	-19.14	4.17	25.42	32.56	26.59
PBIASprom =		-16.48		28.19		
M0025						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-60.00	-60.90	-53.21	-48.12	-37.54	-33.69
5	-60.44	-58.63	-38.64	-36.91	-27.67	-24.89
10	-61.46	-56.02	-47.97	-26.64	-20.65	-20.79
50	-64.15	-57.25	-46.36	2.90	-4.42	-14.24
PBIAS (%) =	-61.51	-58.20	-46.54	-27.19	-22.57	-23.40
PBIASprom =		-55.42		-24.39		
M0033						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-42.70	-40.35	-17.05	-6.86	6.07	15.73
5	-36.93	-36.20	-13.44	-0.71	9.87	11.45
10	-34.20	-30.29	-11.90	3.76	10.67	6.92
50	-30.56	-9.95	-10.37	13.88	10.65	-2.11
PBIAS (%) =	-36.10	-29.20	-13.19	2.52	9.31	8.00
PBIASprom =		-26.16		6.61		
C15						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-43.55	-36.03	-18.41	21.17	38.73	22.22
5	-36.55	-38.16	-18.62	32.53	39.66	21.3
10	-33.91	-40.76	-18.99	36.25	34.29	20.77
50	-29.25	-45.99	-18.76	41.92	16.14	19.79
PBIAS (%) =	-35.82	-40.23	-18.69	32.97	32.2	21.02
PBIASprom =		-31.58		28.73		
P12						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-41.85	-34.24	-18.03	9.91	21.92	22.93
5	-42.14	-31.30	-13.51	17.84	23.88	14.42
10	-42.65	-29.02	-8.91	26.34	29.71	10.79
50	-43.88	-23.44	3.49	45.84	48.90	4.79
PBIAS (%) =	-42.63	-29.50	-9.24	24.98	31.10	13.23
PBIASprom =		-27.12		23.11		
P27						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-40.12	-37.65	-16.17	9.19	22.39	13.53

5	-36.95	-30.53	-14.39	16.44	30.09	7.21
10	-35.49	-26.30	-13.26	20.97	36.59	5.95
50	-33.19	-17.59	-10.21	28.99	51.91	5.47
PBIAS (%) =	-36.44	-28.02	-13.51	18.90	35.25	8.04
PBIASprom =		-25.99			20.73	
C05						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-51.27	-44.29	-28.11	5.19	19.65	22.49
5	-47.57	-42.18	-27.35	15.22	24.09	21.43
10	-42.85	-36.38	-22.25	25.14	30.41	24.67
50	-29.17	-15.15	-4.00	48.47	46.54	35.60
PBIAS (%) =	-42.71	-34.50	-20.43	23.50	30.17	26.05
PBIASprom =		-32.55			26.57	
P09						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-46.03	-40.30	-12.88	31.18	39.75	40.82
5	-45.77	-42.04	-16.43	21.77	29.91	32.12
10	-43.72	-38.78	-15.37	18.62	26.18	29.82
50	-36.80	-24.40	-8.13	14.80	21.84	28.41
PBIAS (%) =	-43.08	-36.38	-13.20	21.59	29.42	32.79
PBIASprom =		-30.89			27.94	
P70						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-34.69	-20.07	-3.73	16.63	13.29	25.45
5	-34.35	-25.43	-18.61	19.14	-1.34	4.94
10	-30.08	-21.12	-19.18	27.23	-3.44	0.71
50	-15.93	-0.50	-11.08	48.83	-2.45	-0.59
PBIAS (%) =	-28.76	-16.78	-13.15	27.96	1.52	7.63
PBIASprom =		-19.56			12.37	
C02						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-33.76	-25.06	-13.80	31.64	33.20	35.42
5	-23.57	-21.10	-14.38	31.31	38.82	34.46
10	-15.20	-14.96	-14.79	32.64	43.88	38.02
50	4.47	3.10	-16.19	36.30	55.67	47.69
PBIAS (%) =	-17.01	-14.50	-14.79	32.97	42.89	38.90
PBIASprom =		-15.44			38.25	
C04						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-45.48	-38.24	-24.25	12.65	10.17	14.04
5	-37.21	-32.91	-17.05	20.72	14.36	9.17
10	-30.46	-26.45	-10.28	24.10	17.85	7.72
50	-14.72	-7.94	5.75	29.32	25.31	7.05
PBIAS (%) =	-31.97	-26.39	-11.46	21.70	16.92	9.49

PBIASprom =		-23.27			16.04	
C14						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-17.32	0.00	2.95	55.74	36.55	21.12
5	-10.31	0.74	11.51	56.50	45.02	17.24
10	-4.47	5.22	20.21	51.55	50.16	17.26
50	8.61	20.06	40.97	38.17	59.79	19.48
E (%) =	-5.87	6.51	18.91	50.49	47.88	18.77
Epro (%) =	6.51			39.05		
P03						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-49.00	-43.91	-28.94	7.95	11.30	12.72
5	-40.74	-36.18	-20.03	12.06	18.21	12.80
10	-35.61	-30.44	-15.76	13.72	22.96	13.25
50	-25.39	-16.42	-9.88	15.96	33.04	13.93
PBIAS (%) =	-37.68	-31.74	-18.65	12.42	21.38	13.17
PBIASprom =	-29.36			15.66		
P08						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-49.80	-43.49	-29.27	10.25	2.52	0.00
5	-41.26	-39.08	-24.38	6.84	4.46	-1.24
10	-35.95	-34.99	-21.12	3.75	7.83	0.38
50	-25.35	-24.13	-14.50	-3.44	17.05	5.63
PBIAS (%) =	-38.09	-35.42	-22.32	4.35	7.97	1.19
PBIASprom =	-31.94			4.50		
P11						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-43.42	-33.46	-20.73	11.98	14.04	16.77
5	-28.26	-22.74	-15.60	23.15	25.97	20.20
10	-17.54	-14.01	-13.44	30.35	34.64	24.41
50	6.68	8.87	-11.12	44.08	52.53	34.60
PBIAS (%) =	-20.63	-15.33	-15.22	27.39	31.79	24.00
PBIASprom =	-17.06			27.73		
P28						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-41.68	-39.27	-26.22	7.55	10.92	2.63
5	-29.97	-30.00	-16.83	16.32	22.67	6.73
10	-21.92	-22.06	-10.55	20.49	28.19	9.50
50	-4.37	-1.28	2.15	26.88	37.59	16.00
PBIAS (%) =	-24.48	-23.15	-12.86	17.81	24.84	8.72
PBIASprom =	-20.17			17.12		
P31						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-27.64	-22.22	-7.46	34.43	37.13	34.48

5	-12.36	-17.74	0.56	41.43	38.36	27.27
10	-2.35	-13.31	5.73	44.68	38.90	26.19
50	18.69	-1.12	15.72	50.30	40.14	27.60
PBIAS (%) =	-5.91	-13.60	3.64	42.71	38.63	28.89
PBIASprom =		-5.29			36.74	
P32						
T	0.5 hr	1 hr	2 hr	6 hr	12 hr	24 hr
2	-14.71	-6.20	-4.33	24.45	20.82	15.38
5	-10.00	-14.34	-2.11	13.62	12.23	6.25
10	-4.24	-16.89	2.64	12.48	12.88	6.85
50	10.01	-18.77	15.04	13.74	18.43	13.52
PBIAS (%) =	-4.73	-14.05	2.81	16.07	16.09	10.50
PBIASprom =		-5.33			14.22	

Elaboración: propia

La estación M0003 para duraciones cortas (30 minutos, 1 hora, 2 horas) las curvas IDF de IMERG V06 están subestimadas, el sesgo es similar para los dos periodos de retorno. Por otra parte, para las duraciones largas (6 horas, 12 horas, 24 horas) existe una sobreestimación de las intensidades satelitales en el periodo de retorno de 50 años y para el periodo de 2 años el sesgo es muy pequeño.

La estación M0006 para duraciones cortas las curvas IDF satelitales están subestimadas, pero el sesgo más grande para estas duraciones se produce en el periodo de retorno de 50 años. Por otra parte, para duraciones altas el mayor sesgo está en el periodo de retorno de 2 años dando una subestimación por parte de IMERG V06, mientras que para el otro periodo de retorno la línea es muy similar.

La estación M0007 para duraciones cortas las curvas IDF satelitales están subestimadas hasta 2 horas, y de 2 horas en adelante están sobreestimados obteniéndose el sesgo más grande para el periodo de retorno de 2 años. Por otra parte, para duraciones altas la curva IDF de los datos de IMERG V06 esta sobreestimados y se produce el mayor sesgo para T=50.

La estación M0025 para duraciones cortas las curvas IDF de IMERG V06 están muy subestimadas, generando un gran sesgo de casi el doble de los datos observados para T=50. Por otra parte, para las largas duraciones en T=50 las líneas parecen juntas y para T=2 existe una subestimación de los datos IMERG V06.

La estación M0033 para duraciones cortas las curvas IDF satelitales están subestimadas, el sesgo más importante se produce para el periodo de retorno de 2 años. Por otra parte, para las duraciones largas, la curva IDF IMERG V06 y la curva observada son similares.

La estación C15 para duraciones cortas las curvas IDF de IMERG V06 están subestimadas, el sesgo es importante para los dos periodos de retorno. Por otra parte, para las duraciones largas, existe una sobreestimación de las intensidades satelitales con un sesgo pequeño en ambos casos y casi despreciable para el periodo de retorno de 2 años.

Las curvas IDF satelitales de las estaciones P12-P27, con las duraciones cortas, estas curvas están subestimadas, el sesgo es parecido para los dos periodos de retorno. Por otra parte, para las duraciones altas con $T=2$ existe una sobreestimación mínima de las intensidades satelitales con un sesgo despreciable y para $T=50$ existe una sobreestimación de las intensidades satelitales con un sesgo mediano.

Las curvas IDF satelitales de las estaciones C05-P09-P70 con las duraciones cortas, estas curvas están subestimadas, el sesgo es más grande para el periodo de retorno de 2 años. Por otra parte, para las duraciones altas con $T=2$ existe una sobreestimación mínima de las intensidades satelitales con un sesgo despreciable y para $T=50$ existe una sobreestimación de las intensidades satelitales con un sesgo mediano.

Las curvas IDF de las últimas 9 estaciones dan resultados parecidos a los que hemos estado teniendo, con una subestimación en las curvas IDF satelitales para duraciones cortas excepto para el periodo de retorno de 50 años con las estaciones C02-C04-C14-P11, y para duraciones largas están sobreestimadas.

En la Costa contamos con dos estaciones M0006 y M0025 curvas IDF para las estaciones costeras tiene un rango de error [-40% : -11%] y [-55% : -24%] respectivamente. Estos rangos significarían que en ambos casos existe una subestimación por parte de IMERG V06 en la zona del Pacífico para todas sus duraciones.

En la Amazonía se tiene la estación M0007 en la que sus curvas IDF presenta el siguiente rango de error [-16% : 28%] lo que resulta en que para duraciones cortas existe un 16% de subestimación de las curvas IDF por parte de IMERG V06 y para duraciones largas existe un 28% de sobreestimación de las curvas IDF satelitales.

El análisis en la Serranía Ecuatoriana comienza en la zona Sur con la estación M0033 que tiene un rango de [-26% : 7], lo que significa que para duraciones cortas existe un 26% de subestimación de las curvas IDF de IMERG V06, para duraciones largas existe un 7% sobreestimación. Otra estación de la Serranía es la M0003 ubicada en la zona central Sur

cuenta con un rango de [-40% : 15], lo que significa que en duraciones cortas existe un 40% de subestimación de las curvas IDF satelitales y para duraciones largas existe un 15% sobreestimación de las curvas IDF de IMERG V06.

Ahora viene el análisis con las estaciones ubicadas en el DMQ por parte del EPMAPS. La estación C15 cuenta de [-32% : 29], lo que significa que para duraciones cortas existe un 32% de subestimación de las curvas IDF de IMERG V06 y para duraciones largas existe un 29% sobreestimación de las intensidades satelitales. La estación P12 para duraciones cortas existe un 27% de subestimación de las curvas IDF de IMERG V06 y para duraciones largas existe un 23% sobreestimación de las intensidades satelitales.

La estación P27 para duraciones cortas existe un 26% de subestimación de las curvas IDF de IMERG V06 y para duraciones largas existe un 21% sobreestimación de las intensidades satelitales. La estación C05 para duraciones cortas existe un 33% de subestimación de las curvas IDF de IMERG V06 y para duraciones largas existe un 27% sobreestimación de las intensidades satelitales. La estación P09 para duraciones cortas existe un 31% de subestimación de las curvas IDF de IMERG V06 y para duraciones largas existe un 28% sobreestimación de las intensidades satelitales.

La estación P70 para duraciones cortas existe un 20% de subestimación de las curvas IDF de IMERG V06 y para duraciones largas existe un 12% sobreestimación de las intensidades satelitales. La estación C02 para duraciones cortas existe un 15% de subestimación de las curvas IDF de IMERG V06 y para duraciones largas existe un 38% sobreestimación de las intensidades satelitales. La estación C04 para duraciones cortas existe un 23% de subestimación de las curvas IDF de IMERG V06 y para duraciones largas existe un 16% sobreestimación de las intensidades satelitales. La estación C14 para duraciones cortas existe un 7% de sobreestimación de las curvas IDF de IMERG V06 y para duraciones largas existe un 39% sobreestimación de las intensidades satelitales.

La estación P03 para duraciones cortas existe un 29% de subestimación de las curvas IDF de IMERG V06 y para duraciones largas existe un 16% sobreestimación de las intensidades satelitales. La estación P08 para duraciones cortas existe un 32% de subestimación de las curvas IDF de IMERG V06 y para duraciones largas existe un 5% sobreestimación de las intensidades satelitales. La estación P11 para duraciones cortas existe un 17% de subestimación de las curvas IDF de IMERG V06 y para duraciones largas existe un 28% sobreestimación de las intensidades satelitales.

La estación P28 para duraciones cortas existe un 20% de subestimación de las curvas IDF de IMERG V06 y para duraciones largas existe un 17% sobreestimación de las

intensidades satelitales. La estación P31 para duraciones cortas existe un 5% de subestimación de las curvas IDF de IMERG V06 y para duraciones largas existe un 37% sobreestimación de las intensidades satelitales. La estación P32 para duraciones cortas existe un 5% de subestimación de las curvas IDF de IMERG V06 y para duraciones largas existe un 14% sobreestimación de las intensidades satelitales.

3.2 Discusión

El producto IMERG V06 para la región Costa cuenta con las estaciones M0006 (120 m s. n. m.) y M0025 (379 m s. n. m.) en los dos casos existe una subestimación del producto IMERG V06 para todas las duraciones, pero el PBIAS es menor en la estación M0006 esto puede deberse a que tiene una menor elevación.

El producto IMERG V06 para la región Sierra cuenta con las estaciones M0003 (3058 m s. n. m.) y M0033 (2160 m s. n. m.) en los dos casos existe una subestimación del producto IMERG V06 para duraciones cortas (0.5, 1, 2 horas) y para duraciones largas (6, 12, 24 horas) hay una sobreestimación del producto IMERG V06, pero el PBIAS es menor en la estación M0033 esto puede deberse a que tiene una menor elevación.

El producto IMERG V06 para la región Amazónica cuenta con la estación M0007 (185 m s. n. m.) en la cual existe una subestimación del producto IMERG V06 para duraciones cortas (0.5, 1, 2 horas) y para duraciones largas (6, 12, 24 horas) hay una sobreestimación del producto IMERG V06, en este caso la diferencia es que existe una mayor sobreestimación (28) que una subestimación (-16).

El producto IMERG V06 para los pixeles de EPMAPS en el DMQ, existe una subestimación del producto IMERG V06 para duraciones cortas (0.5, 1, 2 horas) y para duraciones largas (6, 12, 24 horas) hay una sobreestimación del producto IMERG V06. Las estaciones , P27 (3551 m s. n. m.), P70 (2903 m s. n. m.) y P32 (3330 m s. n. m.) son las de mejores resultados cada una en su respectivo píxel. La relación entre estas estaciones es que son las más cercanas al Oeste del píxel, es decir aledañas a la región Costa porque por elevación tienen una altura mayor a 2900 m s. n. m. pero hay estaciones que sobrepasan este nivel, entonces en los casos de EPMAPS no es cuestión de elevación, sino es cuestión de cercanía a la región Costa para un mejor rendimiento.

En Kyaw et al., (2022) los resultados de las curvas IDF arrojaron que todos los productos satelitales subestimaron las curvas IDF diarias observadas para distintos periodos de retorno en la zona costera del Aeropuerto Internacional de Yangon (YIA), excepto el producto IMERG donde solo fue el 19,5 al 29,1%. En general, las diferencias más significativas se registraron en las precipitaciones de 1,5 h a 6 h de duración, lo que

coincide con nuestro estudio en donde se producía un sesgo más grandes a cortas duraciones (0.5, 1, 2 horas) . En resumen, IMERG fue el mejor para generar curvas IDF en YIA debido a que la sobreestimación de IMERG fue mucho menor que las subestimaciones de otros productos satelitales.

Por otro lado, Lau & Behrangi, (2022), en Arizona-Estados Unidos y Feldbach-Austria los valores negativos de las curvas IDF significan que IMERG subestiman las intensidades de precipitación observada para las duraciones más cortas. Las IDF de IMERG ajustadas con Gumbel y MLE tienden a producir los sesgos más bajos. Los resultados muestran que para duraciones cortas (0,5 y 1 h), existe una subestimación sistemática de las intensidades de lluvia en casi todas las estaciones y que para duraciones largas (2 y 6 h) ocurre lo contrario, con una sobreestimación generalizada de las intensidades de lluvia. Algo muy similar a nuestros resultados que para duraciones cortas (0.5, 1, 2 horas) se subestima el IMERG V06 y para duraciones largas (6, 12, 24 horas) se produce una sobreestimación.

Según Rodríguez-Sandoval et al., (2023), para las estaciones pluviométricas en Colombia en los Andes y el Pacífico, las curvas IDF son bastante similares, con errores bajos que se hacen grandes para las duraciones de 2 y 6 h. La estación de pluviómetros en la región del Caribe muestra uno de los resultados más pobres en general. Existen importantes subestimaciones de las intensidades para duraciones cortas (0,5 y 1 h) con un máximo de -55%. Para duraciones largas, las IDF observadas y simuladas son similares, con errores relativos máximos de $\pm 20\%$. Algo muy similar a nuestros resultados que para duraciones cortas (0.5, 1, 2 horas) se subestima el IMERG V06 y para duraciones largas (6, 12, 24 horas) se produce una sobreestimación.

En este estudio, las curvas IDF de IMERG V06 presentan una subestimación para duraciones cortas (0.5, 1, 2 horas) y hay sobrestimación de las curvas IDF IMERG V06 para duraciones largas (6, 12, 24 horas). Lo que coincide con los resultado de Lau & Behrangi, (2022), al igual que con Rodríguez-Sandoval et al., (2023) donde es mucho más similar para duraciones largas que para las duraciones cortas. Ciertos casos graficados del estudio en los Andes y el Caribe revelan una sobreestimación en duraciones cortas (0.5, 1 hora) lo que no coincide con la mayoría de las estaciones de nuestro estudio. Existe un periodo de transición el cual comprende el periodo entre 2 a 6 horas donde existe un cambio por lo general de subestimación a sobreestimación de IMERG V06.

3.3 Conclusiones

- El desempeño del producto IMERG V06 en su versión final con escala temporal de 30 minutos, resolución espacial de 0.1° en el periodo comprendido entre los años

2000-2021 es bastante variable y depende de las características fisiográficas y climatológicas de los lugares donde se ubican las estaciones. Sin embargo, el análisis agregado para todas las estaciones en las tres regiones de Ecuador muestra que en la mayoría de las estaciones para las curvas IDF obtenidas en base a IMERG, existen subestimaciones de las intensidades de lluvia del producto IMERG V06 para duraciones cortas (30 minutos, 1 h, 2h), entre el periodo de 2h a 6h horas existe el cambio de subestimación a sobreestimación y sobreestimaciones para las duraciones largas (6, 12 y 24 h), con errores relativos en los rangos [-55%, +7%] y [-24%, +38%] respectivamente.

- Los mejores resultados de las métricas de validación se obtienen para las duraciones más largas, siendo muy similares los valores observados y los IMERG V06. El coeficiente de correlación de Pearson, RMSE, PBIAS y NSE resalta que los mejores resultados se obtienen para las estaciones de EPMAPS. Si bien esto podría estar asociado con la alta cantidad de datos y poca cantidad de porcentaje de vacíos. Las comparaciones de las curvas IDF observadas de EPMAPS y IMERG con las observadas refuerzan estos hallazgos.
- Las estaciones de INAMHI se encuentra localizadas en zonas estratégicas del Ecuador, la estación con mayor similitud de curvas IDF observadas e IMERG V06 por parte de INAMHI es la M0006, perteneciente a la región Costa. Le sigue la estación M0025 y la M0003 que tuvieron resultados parecidos la primera nuevamente se encuentra en la Costa y la última es de la zona sur de la Serranía Ecuatoriana. En la Amazonia la estación M0007 las similitudes son bajas ya que es el penúltimo rango más grande de error, y los peores resultados por parte del INAMHI fue la estación M0003 ubicada en la zona alta de la Serranía.
- Las mejores concordancias entre las intensidades de las curvas IDF derivadas del IMERG V06 y las del EPMAPS se dan para la estación pluviométrica P32 ubicada en el DMQ Mindo Bajo. Para otras estaciones, los resultados son inferiores, con mayores errores relativos en las intensidades de lluvia.
- Las funciones de distribución que predominaron con mejores ajustes fueron las de Pearson III y Log Pearson III.
- Nuestros resultados contribuyen a una mejor comprensión y un posible análisis del sesgo obtenido respecto a la utilización de IMERG V06 para derivar curvas IDF. El uso de IMERG V06 aplicado una corrección se podría convertir en una alternativa

importante para obtener las curvas IDF en regiones donde exista un registro insatisfactorio de datos o simplemente no exista.

3.4 Recomendaciones

- Se debe revisar las unidades de los datos proporcionados por IMERG V06, y leer detenidamente sus propiedades ya que es importante saber las unidades iniciales mm/h (intensidad), para cambiarlas o dejarlas para el análisis según se necesite en nuestro caso las cambiamos por mm (precipitación).
- Tener un cuidado especial en la temporalidad de los datos debido a que los datos están medidos en la zona horaria (0:00 UTC), es decir el origen, pero Ecuador se encuentra en la zona horaria -5:00 (BOGOTA/QUITO UTC). Por esta razón hay que quitar 5 horas a la hora del satélite, obviamente estas 5 horas hay que transformarlas a minutos que son las unidades de tiempo (30 minutos) en las que viene los datos IMERG V06.
- Si la información obtenida en este estudio, se la quiere usar para otro lugar cercano, hay que ser cuidadosos al extrapolar los resultados.
- Este estudio puede repetirse en algún futuro cuando los datos observados y satelitales estén disponibles por un período más largo, cuando exista más información por parte del INAMHI y EPMAPS.
- Realizar una verificación manual en el proceso de agregación de las series para estar seguros de que se haya hecho de manera correcta este procedimiento.
- Utilizar este estudio de base para realizar una corrección de sesgo para las curvas IDF IMERGV06 y las observadas.

4 REFERENCIAS BIBLIOGRÁFICAS

Acosta Castellanos, P. M., & Sierra Aponte, L. X. (2013). Evaluación de métodos de construcción de curvas IDF a partir de distribuciones de probabilidad y parámetros de ajuste. *REVISTA FACULTAD DE INGENIERÍA*, 22(35), 25. <https://doi.org/10.19053/01211129.2512>

Apolo Cumbicos, J. C., & Taco Taco, B. B. (2022). *Desarrollo de un sistema web basado en una arquitectura de microservicios para la generación de reportes de datos hidrometeorológicos de las estaciones convencionales del INAMHI*.

Boers, N., Bookhagen, B., Marwan, N., Kurths, J., & Marengo, J. (2013). Complex networks identify spatial patterns of extreme rainfall events of the South American Monsoon

System: SPATIAL PATTERNS OF EXTREME RAINFALL. *Geophysical Research Letters*, 40(16), 4386-4392. <https://doi.org/10.1002/grl.50681>

Boluwade, A., Stadnyk, T., Fortin, V., & Roy, G. (2017). Assimilation of precipitation Estimates from the Integrated Multisatellite Retrievals for GPM (IMERG, early Run) in the Canadian Precipitation Analysis (CaPA). *Journal of Hydrology: Regional Studies*, 14, 10-22. <https://doi.org/10.1016/j.ejrh.2017.10.005>

Buytaert, W., Celleri, R., Willems, P., Bièvre, B. D., & Wyseure, G. (2006). Spatial and temporal rainfall variability in mountainous areas: A case study from the south Ecuadorian Andes. *Journal of Hydrology*, 329(3-4), 413-421. <https://doi.org/10.1016/j.jhydrol.2006.02.031>

Campozano, L., Céleri, R., Trachte, K., Bendix, J., & Samaniego, E. (2016). Rainfall and Cloud Dynamics in the Andes: A Southern Ecuador Case Study. *Advances in Meteorology*, 2016, 1-15. <https://doi.org/10.1155/2016/3192765>

Cardoso De Salis, H. H., Monteiro Da Costa, A., Moreira Vianna, J. H., Azeneth Schuler, M., Künne, A., Sanches Fernandes, L. F., & Leal Pacheco, F. A. (2019). Hydrologic Modeling for Sustainable Water Resources Management in Urbanized Karst Areas. *International Journal of Environmental Research and Public Health*, 16(14), 2542. <https://doi.org/10.3390/ijerph16142542>

Chavoshi, S. (2009). Development of L-Moment Based Models for Extreme Flood Events. *Malaysian Journal of Mathematical Sciences*.

Chen, F., & Li, X. (2016). Evaluation of IMERG and TRMM 3B43 Monthly Precipitation Products over Mainland China. *Remote Sensing*, 8(6), 472. <https://doi.org/10.3390/rs8060472>

Chow, V. T., Maidment, D., & Mays, L. (2000). *Hidrología aplicada*. McGraw-Hill.

De Carvalho, L. M. V., & Cavalcanti, I. F. A. (2016). The South American Monsoon System (SAMS). En L. M. V. De Carvalho & C. Jones (Eds.), *The Monsoons and Climate Change* (pp. 121-148). Springer International Publishing. https://doi.org/10.1007/978-3-319-21650-8_6

Fitzgerald, D. L. (2007). *Estimation of Point Rainfall Frequencies*.

Golmohammadi, G., Prasher, S., Madani, A., & Rudra, R. (2014). Evaluating Three Hydrological Distributed Watershed Models: MIKE-SHE, APEX, SWAT. *Hydrology*, 1(1), 20-39. <https://doi.org/10.3390/hydrology1010020>

Hosking, J. R. M., Wallis, J. R., & Wood, E. F. (1985). *Estimation of the Generalized Extreme- Value Distribution by the Method of Probability-Weighted Moments*. 27(3).

Hou, A. Y., Kakar, R. K., Neeck, S., Azarbarzin, A. A., Kummerow, C. D., Kojima, M., Oki, R., Nakamura, K., & Iguchi, T. (2014). The Global Precipitation Measurement Mission. *Bulletin of the American Meteorological Society*, 95(5), 701-722. <https://doi.org/10.1175/BAMS-D-13-00164.1>

Huffman, G. J., Bolvin, D. T., Braithwaite, D., Hsu, K., Joyce, R., Kidd, C., Nelkin, E. J., Sorooshian, S., Tan, J., & Pingping, X. (2017). Medición de la precipitación global (GPM) de la NASA, recuperaciones multisatélite integradas para GPM. *Versión del documento de base teórica del algoritmo (ATBD)*, 4.6, 32.

Huffman, G. J., Bolvin, D. T., Nelkin, E. J., & Tan, J. (2019). *Integrated Multi-satellite Retrievals for GPM (IMERG) Technical Documentation*. 77.

Instituto de Meteorología. Cuba, & Planos Gutiérrez, E. O. (2015). Influencia de las grandes precipitaciones en la garantía de entrega de agua anual. *Aqua-LAC*, 7(2), 38-48. <https://doi.org/10.29104/phi-aqualac/2015-v7-2-05>

Koutsoyiannis, D. (2004). Statistics of extremes and estimation of extreme rainfall: I. Theoretical investigation / Statistiques de valeurs extrêmes et estimation de précipitations extrêmes: I. Recherche théorique. *Hydrological Sciences Journal*, 49(4), 3. <https://doi.org/10.1623/hysj.49.4.575.54430>

Kyaw, A. K., Shahid, S., & Wang, X. (2022). Remote Sensing for Development of Rainfall Intensity–Duration–Frequency Curves at Ungauged Locations of Yangon, Myanmar. *Water*, 14(11), 1699. <https://doi.org/10.3390/w14111699>

Lalinde, J. D. H., Castro, F. E., & Rodríguez, J. E. (2018). *Sobre el uso adecuado del coeficiente de correlación de Pearson: Definición, propiedades y suposiciones*. 37.

Lau, A., & Behrangi, A. (2022). Understanding Intensity–Duration–Frequency (IDF) Curves Using IMERG Sub-Hourly Precipitation against Dense Gauge Networks. *Remote Sensing*, 14(19), 5032. <https://doi.org/10.3390/rs14195032>

Loyola Cueva, E. K., & Salazar Hinojosa, V. H. (2022). *CORRELACIÓN ESTADÍSTICA DE ESTACIONES METEOROLÓGICAS CONVENCIONALES Y AUTOMÁTICAS DURANTE EL PERIODO 2014-2021*.

Manz, B., Páez-Bimos, S., Horna, N., Buytaert, W., Ochoa-Tocachi, B., Lavado-Casimiro, W., & Willems, B. (2017). Comparative Ground Validation of IMERG and TMPA at Variable Spatiotemporal Scales in the Tropical Andes. *Journal of Hydrometeorology*, 18(9), 2469-2489. <https://doi.org/10.1175/JHM-D-16-0277.1>

Mehta, A., & McCartney, S. (2020). *Aplicaciones de Reanálisis de GPM IMERG1 para Evaluar Períodos de Sequedad y Humedad Extrema*.

Mélice, J.-L., & Reason, C. J. C. (2007). Return period of extreme rainfall at George, South Africa. *South African Journal of Science*.

Méndez Rivas, R. A. (2018). *Productos de precipitación satelital de alta resolución espacial y temporal en zonas de topografía compleja* [Magíster en Ciencias de la Ingeniería]. <https://doi.org/10.7764/tesisUC/ING/21480>

Morán, W. C. (2003). *Hidrología para estudiantes de ingeniería civil* (Segunda).

Nash, J. E., & Sutcliffe, J. V. (1970). River flow forecasting through conceptual models part I — A discussion of principles. *Journal of Hydrology*, 10(3), 282-290. [https://doi.org/10.1016/0022-1694\(70\)90255-6](https://doi.org/10.1016/0022-1694(70)90255-6)

Negrón Baez, P. A. (2014). *Redes neuronales sigmoidal con algoritmo LM para pronóstico de tendencia del precio de las acciones del IPSA*.

Ovando, G., Sayago, S., & Bellini, Y. (2018). *Evaluación del desempeño de productos satelitales para estimar precipitación en Córdoba (Argentina)*.

Parsons, M. A., Duerr, R., & Minster, J.-B. (2010). Data Citation and Peer Review. *Eos, Transactions American Geophysical Union*, 91(34), 297. <https://doi.org/10.1029/2010EO340001>

Pourrut, P. (Ed.). (1995). *El agua en el Ecuador: Clima, precipitaciones, escorrentía*. Colegio de Geógrafos del Ecuador : Corporación Editora Nacional.

Pulgarín Dávila, E. G. (2009). *Fórmulas regionales para la estimación de curvas intensidad-frecuenciaduración basadas en las propiedades de escala de la lluvia*.

Recalde-Coronel, G. C., Zaitchik, B., & Pan, W. K. (2020). Madden–Julian oscillation influence on sub-seasonal rainfall variability on the west of South America. *Climate Dynamics*, 54(3-4), 2167-2185. <https://doi.org/10.1007/s00382-019-05107-2>

Rew, R., & Davis, G. (1990). NetCDF: An interface for scientific data access. *IEEE Computer Graphics and Applications*, 10(4), 76-82. <https://doi.org/10.1109/38.56302>

Roberto Pizarro Tapia, Claudia Sangüesa Pool, Per Bjarne Bro, Ben Ingram, Mauricio Vera Camiroaga, Carlos Vallejos Carrera, Carolina Morales Calderón, Claudio Olivares Santelices, Francisco Balocchi Contreras, Roberto Fuentes Lagos, Romina Mendoza Mendoza, Alejandro Abarza Martínez, & Rodrigo Valdés Pineda. (2012). *Identificación de prioridades para la gestión nacional de las especies exóticas invasoras*. Comité Nacional de Especies Exóticas Invasoras.

Rodríguez-Sandoval, E. A., García-Echeverri, C., González-Ramírez, A. M., Sandoval-Barrera, J. A., Patarroyo-González, M. E., Agudelo D., D. E., & Roldán N., M. A. (2023). Evaluating the IMERG precipitation satellite product to derive intensity-duration-frequency curves in Colombia. *Revista Facultad de Ingeniería Universidad de Antioquia*. <https://doi.org/10.17533/udea.redin.20230212>

Ruiz-Hernández, J.-C., Condom, T., Ribstein, P., Le Moine, N., Espinoza, J.-C., Junquas, C., Villacís, M., Vera, A., Muñoz, T., Maisincho, L., Campozano, L., Rabatel, A., & Sicart, J.-E. (2021). Spatial variability of diurnal to seasonal cycles of precipitation from a high-altitude equatorial Andean valley to the Amazon Basin. *Journal of Hydrology: Regional Studies*, 38, 100924. <https://doi.org/10.1016/j.ejrh.2021.100924>

Salio, P., Hobouchian, M. P., García Skabar, Y., & Vila, D. (2015). Evaluation of high-resolution satellite precipitation estimates over southern South America using a dense rain gauge network. *Atmospheric Research*, 163, 146-161. <https://doi.org/10.1016/j.atmosres.2014.11.017>

Shen, Y., Xiong, A., Wang, Y., & Xie, P. (2010). Performance of high-resolution satellite precipitation products over China. *Journal of Geophysical Research*, 115(D2), D02114. <https://doi.org/10.1029/2009JD012097>

Singh, V. P., & Chow, V. T. (Eds.). (2017). *Handbook of applied hydrology* (Second edition). McGraw-Hill Education.

Soro, G. E., Goula Bi, T. A., Kouassi, F. W., & Srohourou, B. (2010). Update of Intensity-Duration-Frequency Curves for Precipitation of Short Durations in Tropical Area of West Africa (Cote D'ivoire). *Journal of Applied Sciences*, 10(9), 704-715. <https://doi.org/10.3923/jas.2010.704.715>

Stedinger, J. R. (1980). Fitting log normal distributions to hydrologic data. *Water Resources Research*, 16(3), 481-490. <https://doi.org/10.1029/WR016i003p00481>

U.S. Water Resources Council. (1981). *Guidelines for determining Flood Flow Frequency*.

Varas C., E., & Bois, P. (1998). *Hidrología probabilística* (1a. ed). Eds. Univ. Católica de Chile.

Vera, O. O., Aquíñigo, N. M., Marcacuzco, A. M., & Vásquez, A. (2016). *MORFOMETRÍA ADIMENSIONAL DE SISTEMAS HIDROLÓGICOS ALTOANDINOS Y GENERACIÓN DE INFORMACIÓN HIDROLÓGICA*.

Vuille, M., Bradley, R. S., & Keimig, F. (2000). Climate Variability in the Andes of Ecuador and Its Relation to Tropical Pacific and Atlantic Sea Surface Temperature Anomalies. *Journal of Climate*, 13(14), 2520-2535. [https://doi.org/10.1175/1520-0442\(2000\)013<2520:CVITAO>2.0.CO;2](https://doi.org/10.1175/1520-0442(2000)013<2520:CVITAO>2.0.CO;2)

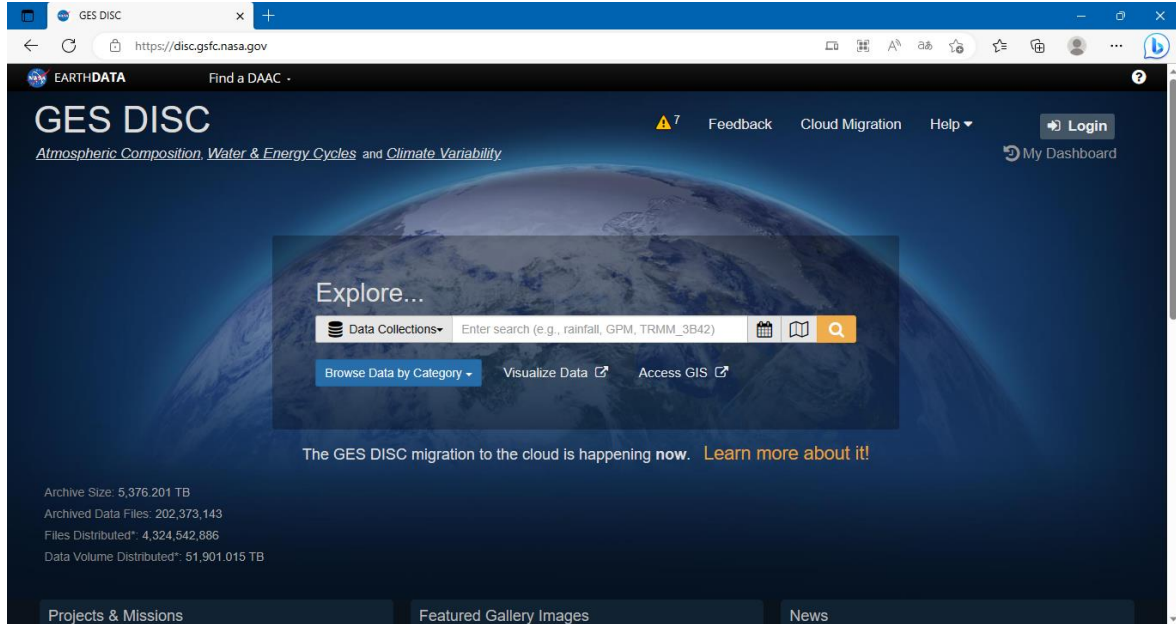
Zozaya, A. (2019). *Método de los Momentos*. <https://doi.org/10.13140/RG.2.2.22925.49120>

5 ANEXOS

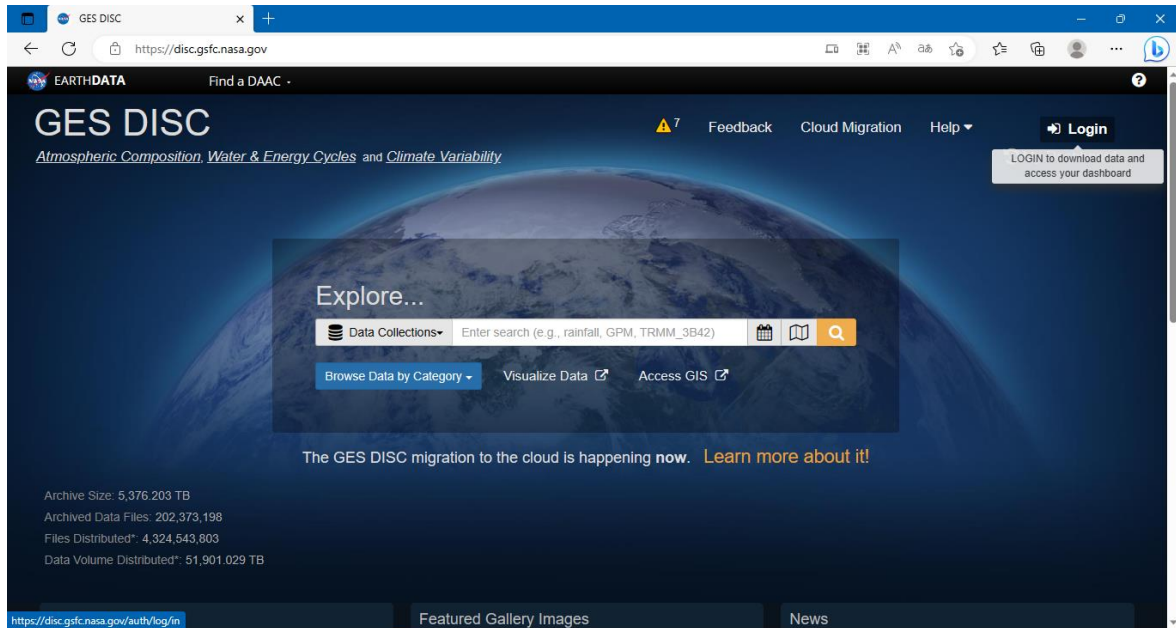
Anexo 1. Manual de descarga del producto IMERG V06

OBTENER LA LISTA DE DESCARGA

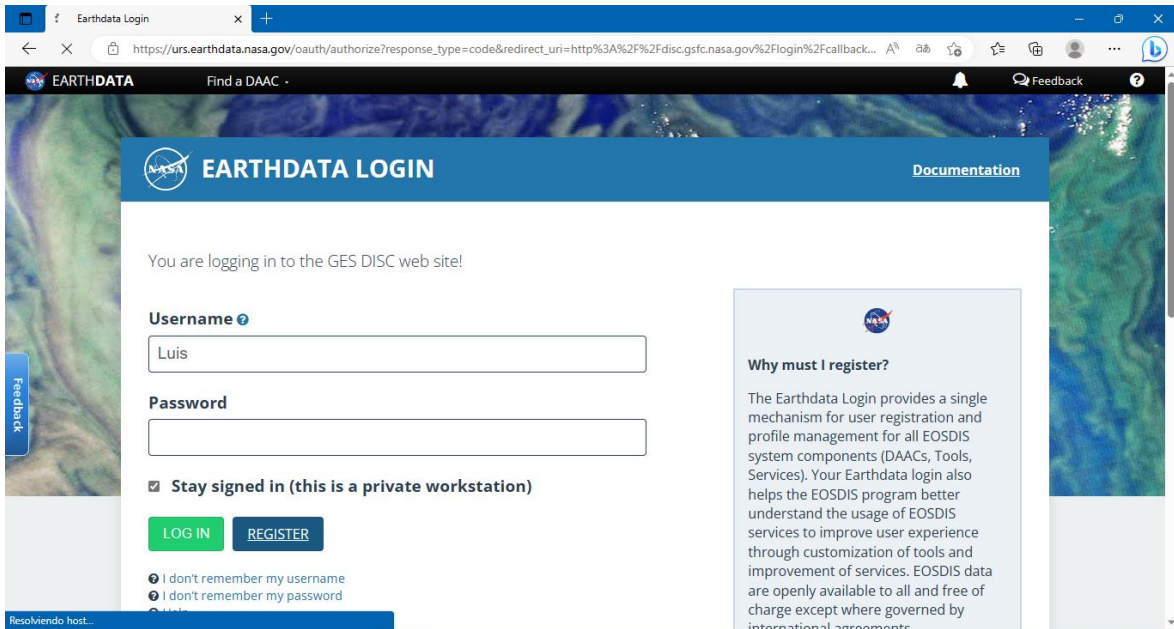
1. Ir a [GES DISC \(nasa.gov\)](https://disc.gsfc.nasa.gov).



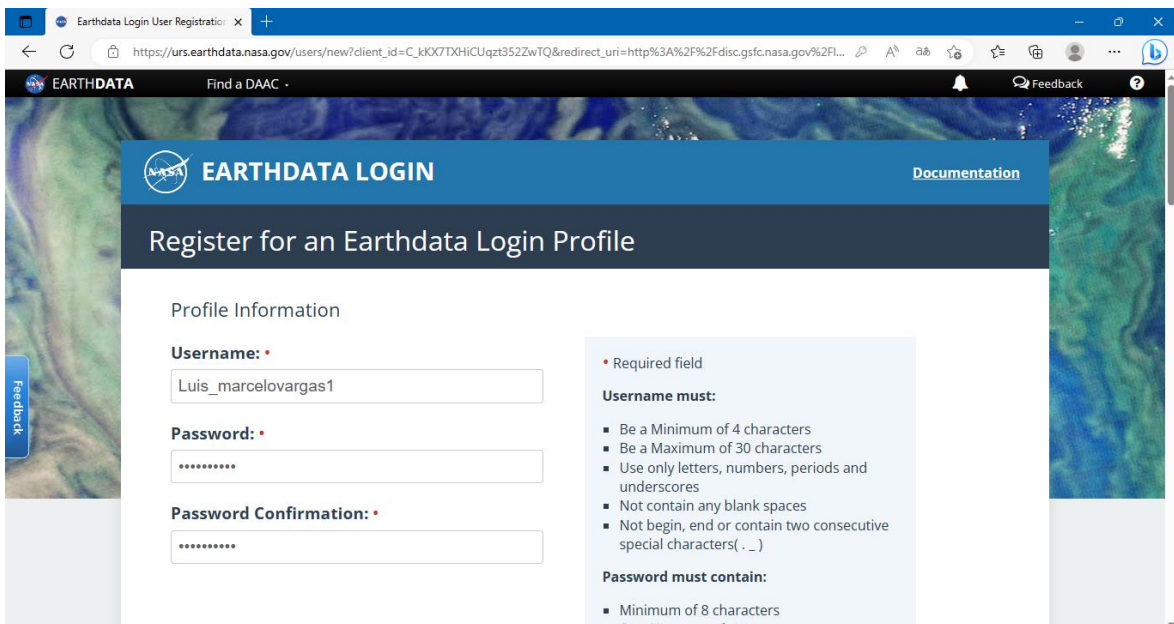
2. Clic en LOGIN.



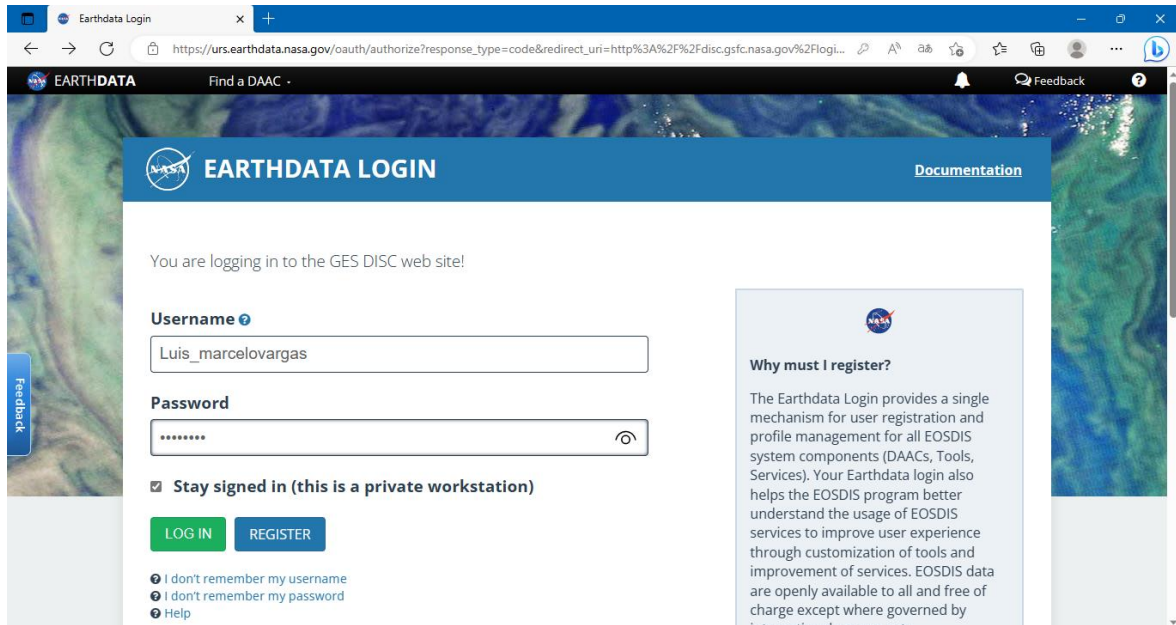
3. En caso de no tener cuenta GESDISC, registrarse REGISTER.



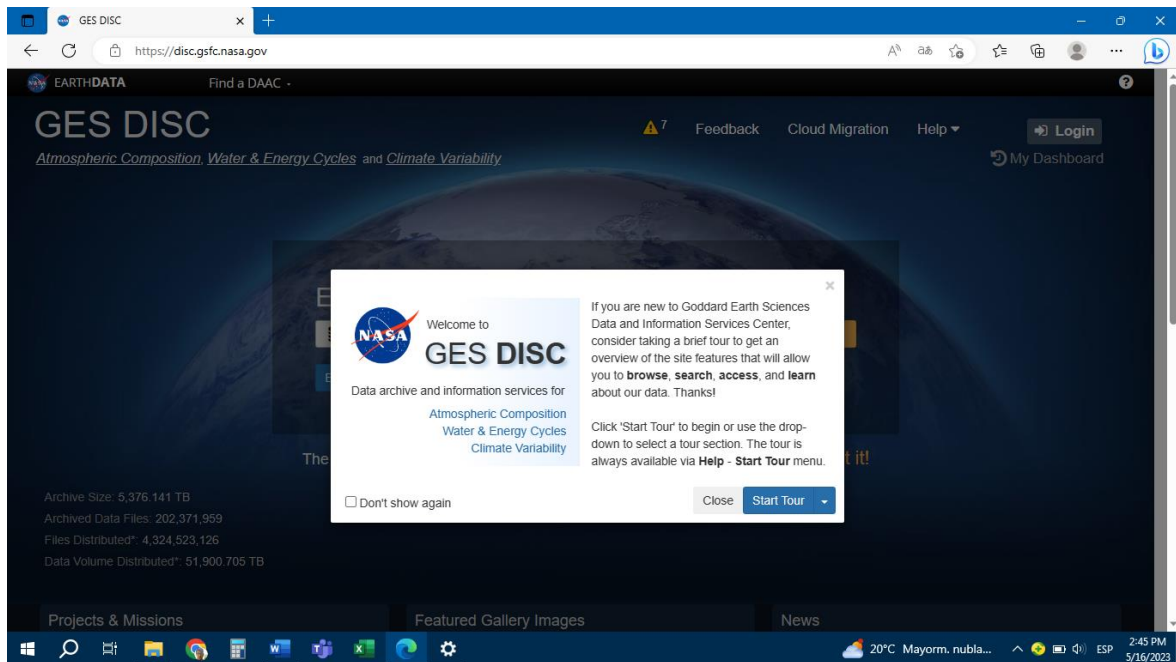
4. Llenar los datos de requeridos y aceptar los términos.



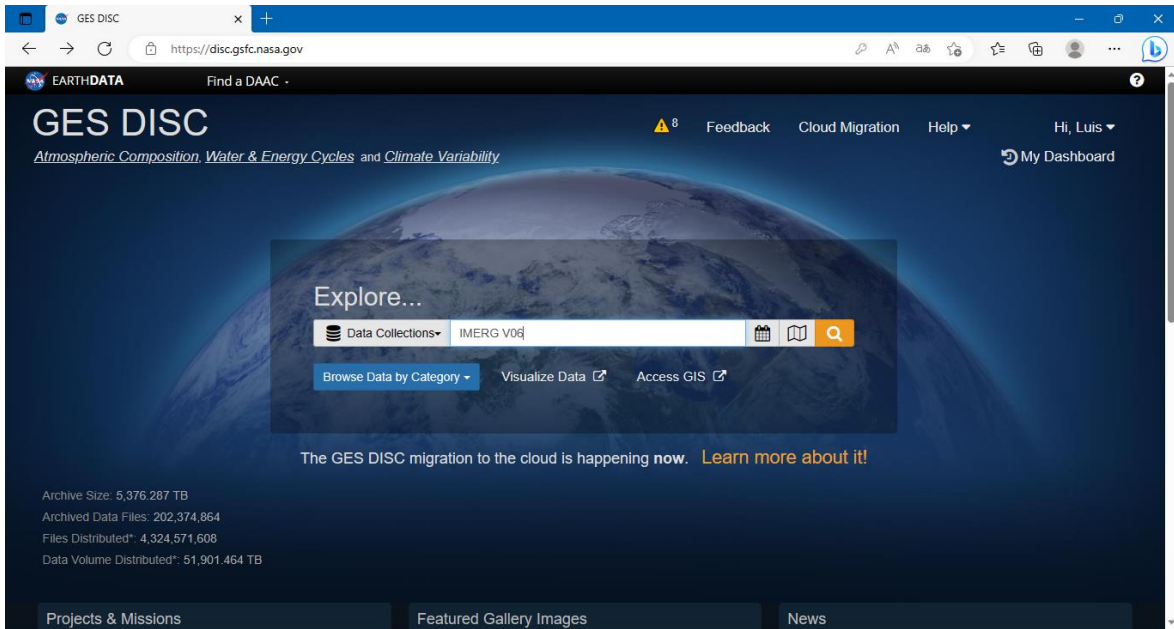
5. Entrar a la cuenta GES DISC.



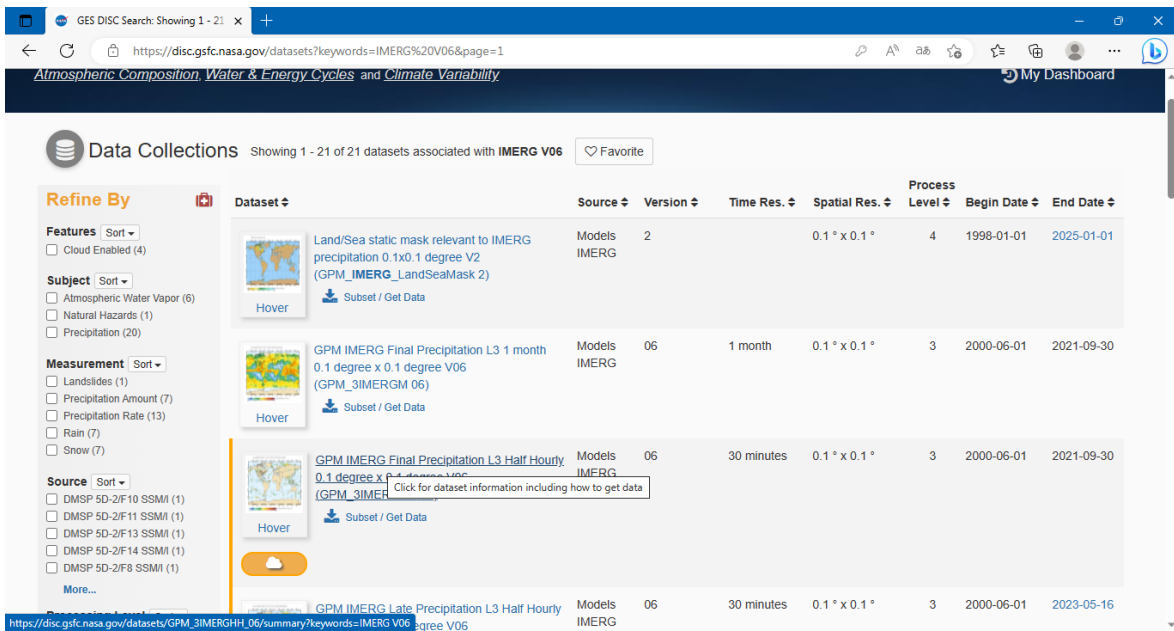
6. En caso de ser principiante es una buena idea seguir el tour que propone el GES DISC, para familiarizarse con la página.



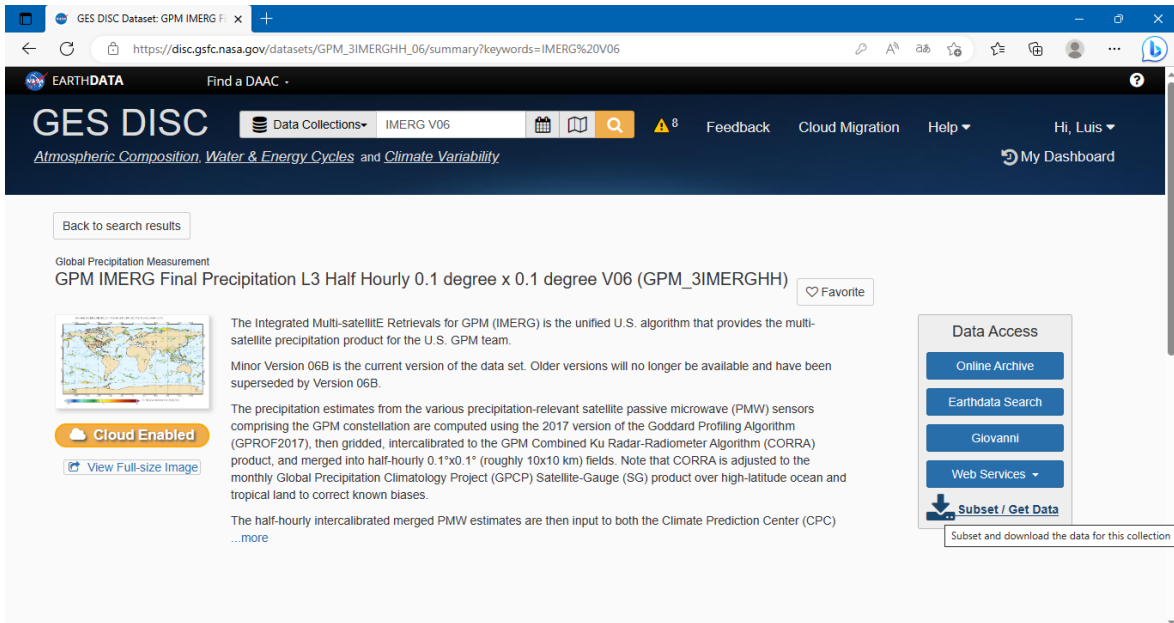
7. En el explorador del GES DISC escribir IMERG V06 y buscar.



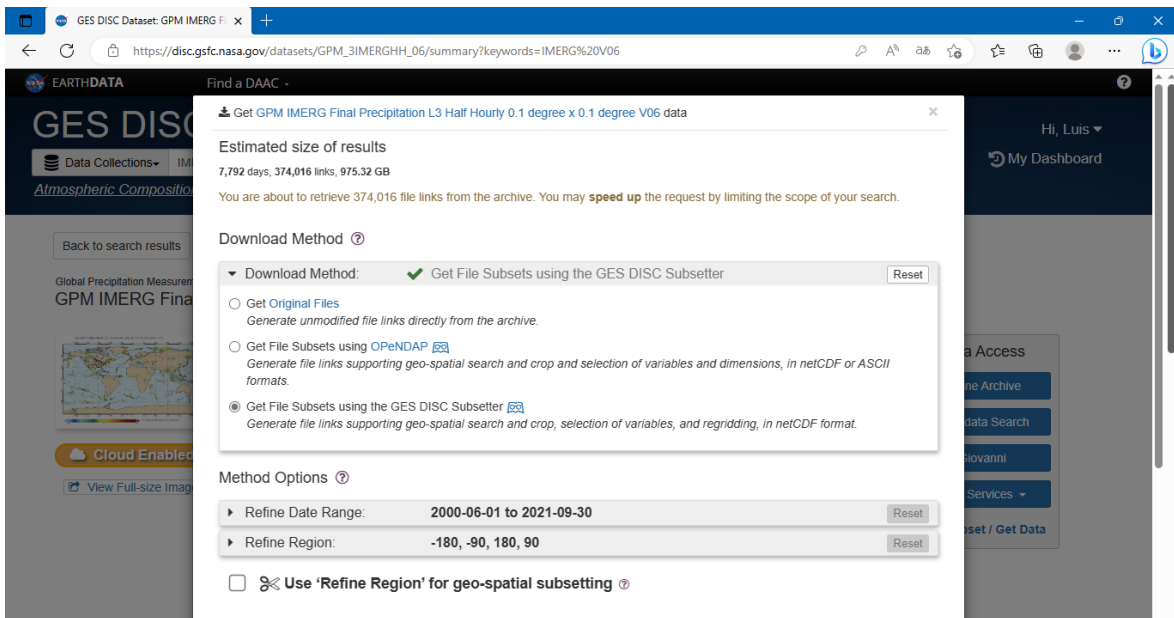
8. Elegir la opción GPM IMERG Final Precipitation L3 Half Hourly 0.1-degree x 0.1-degree V06 (GPM_3IMERGHH).



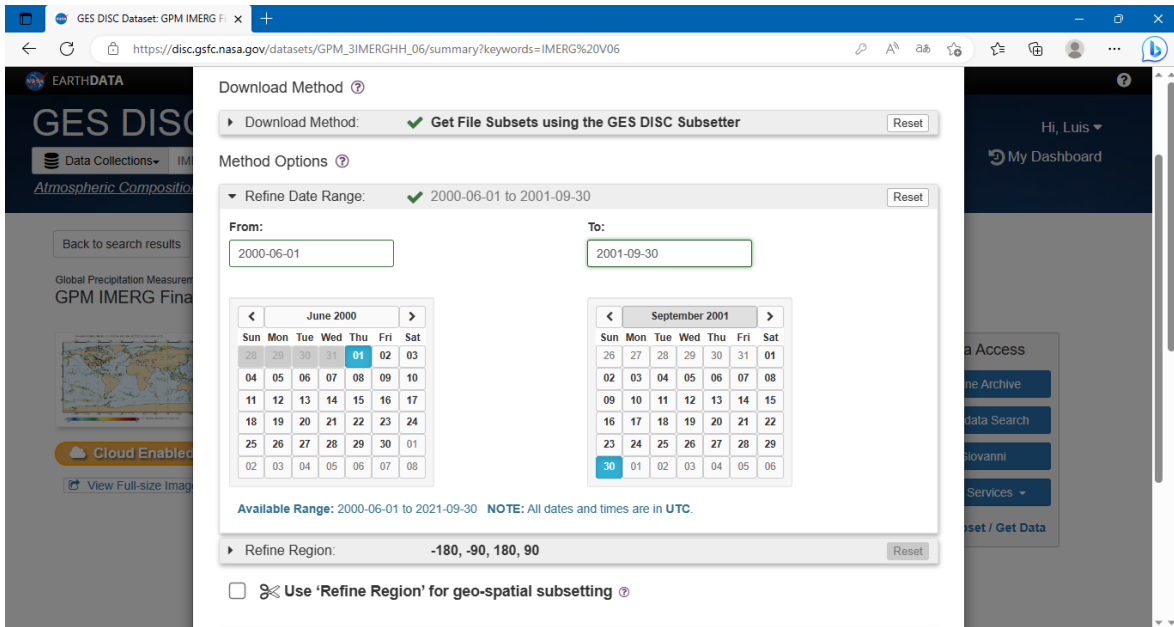
9. Clic en Subset / Get Data.



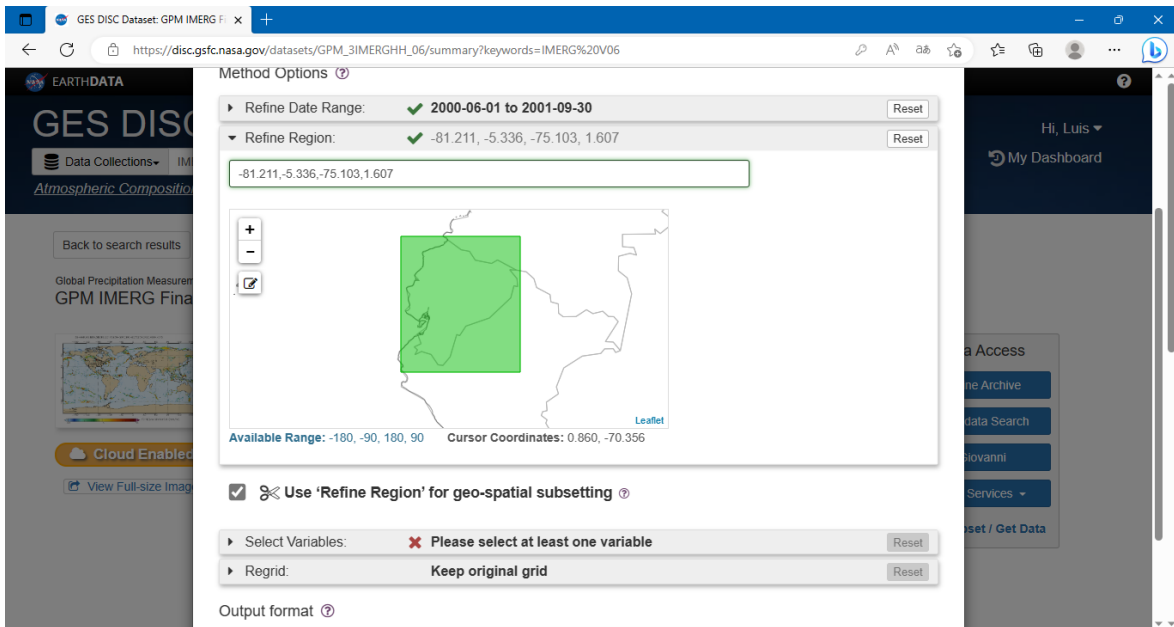
10. En el siguiente cuadro de dialogo, elegir para DOWNLOAD METHOD Get File Subsets using the GES DISC Subsetter.



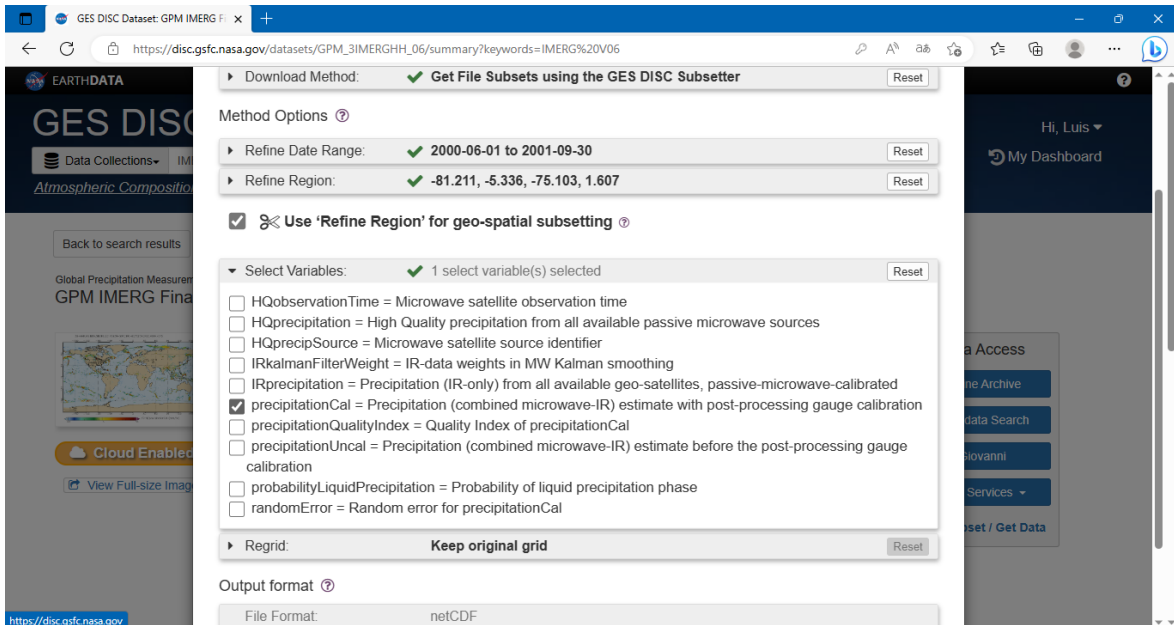
11. Establecer el rango de descarga de datos en Refine Range Date.



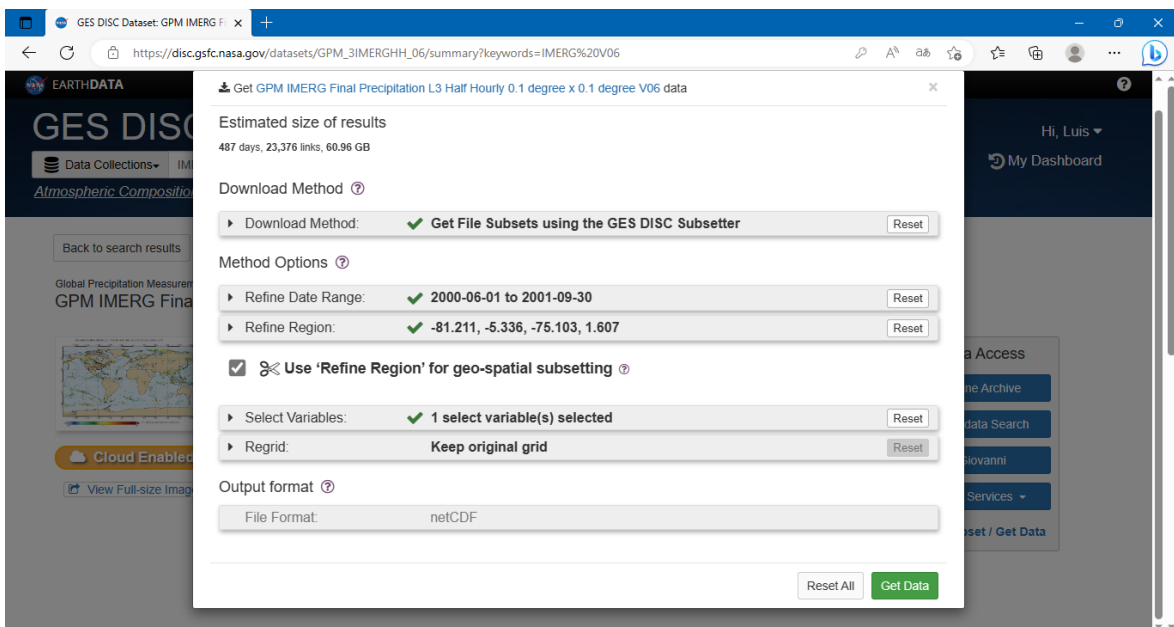
12. Escribir las coordenadas de la ubicación para los datos en Refine Region, y elegir la opción Use “Refine Region” for geo-spatial sub-setting



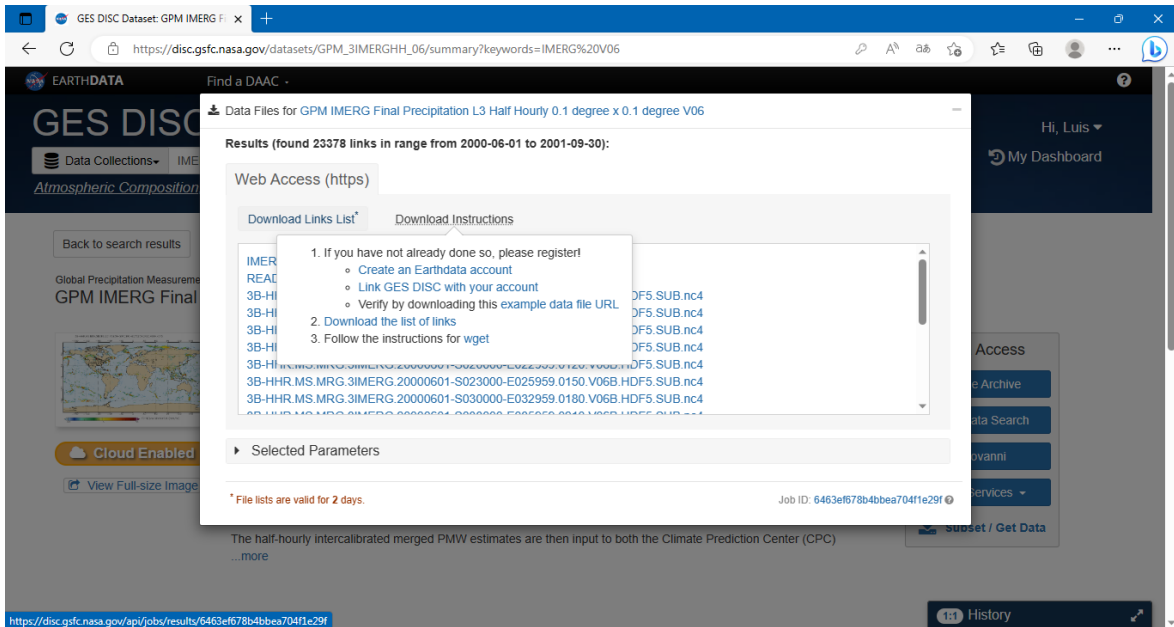
13. Elegir a precipitaciónCAL como variable.



14. Seleccionar Keep Original grid, además el formato netCDF y clic en Get Data.

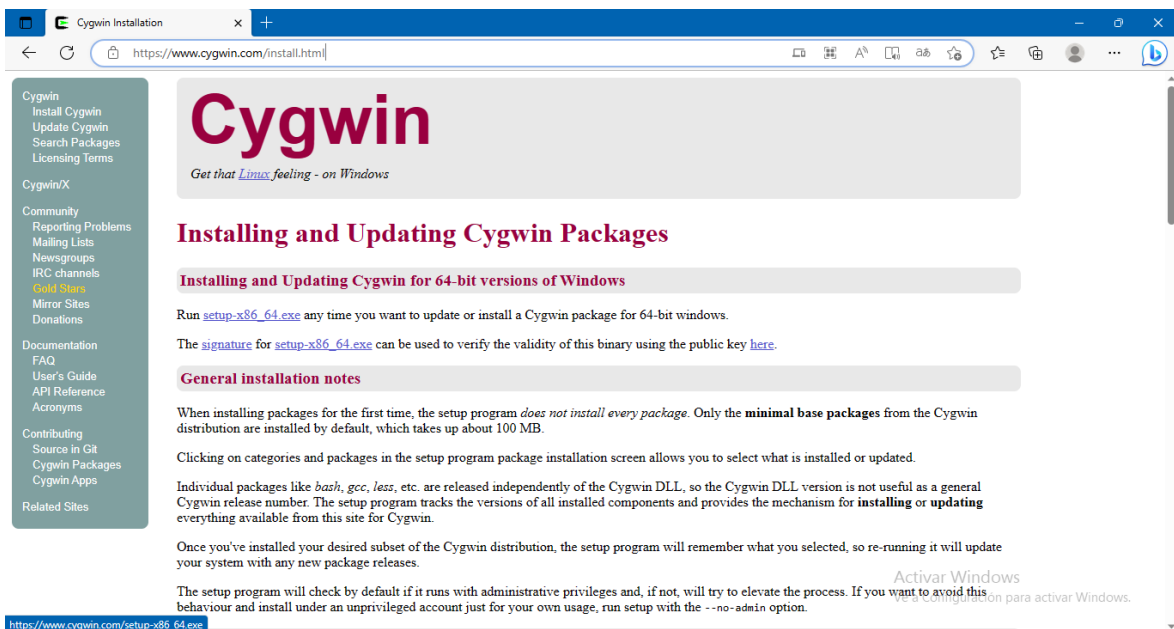


15. Esperar a que se cargue por completo la lista de descarga, cuando se encuentre al 100%, seleccionar Download List

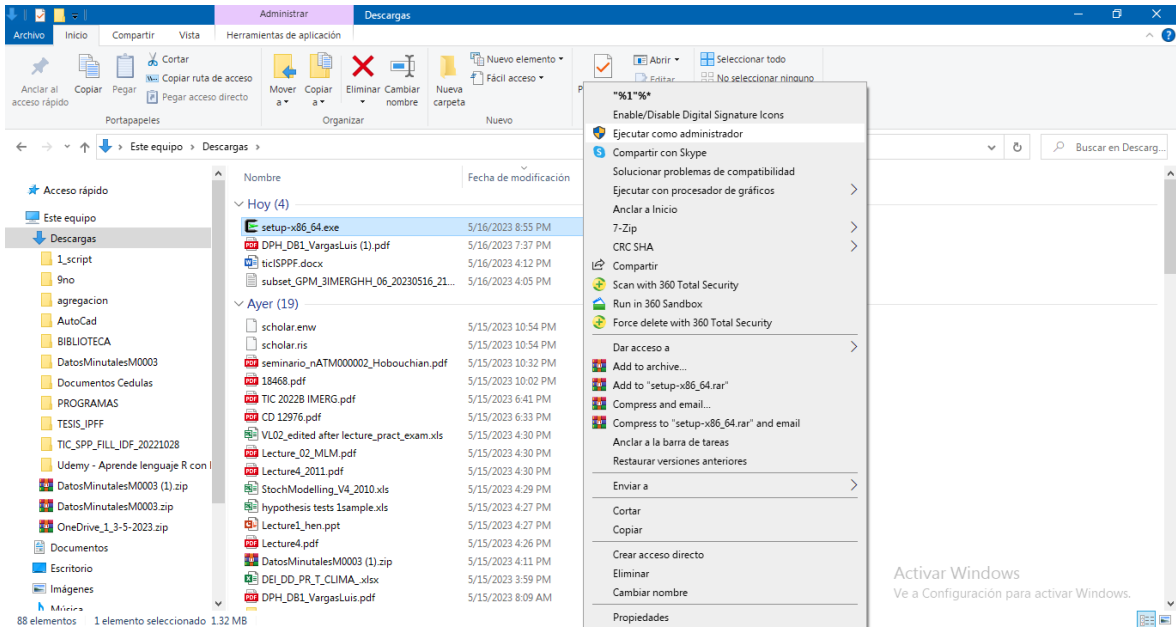


DESCARGA DEL CYGWIN

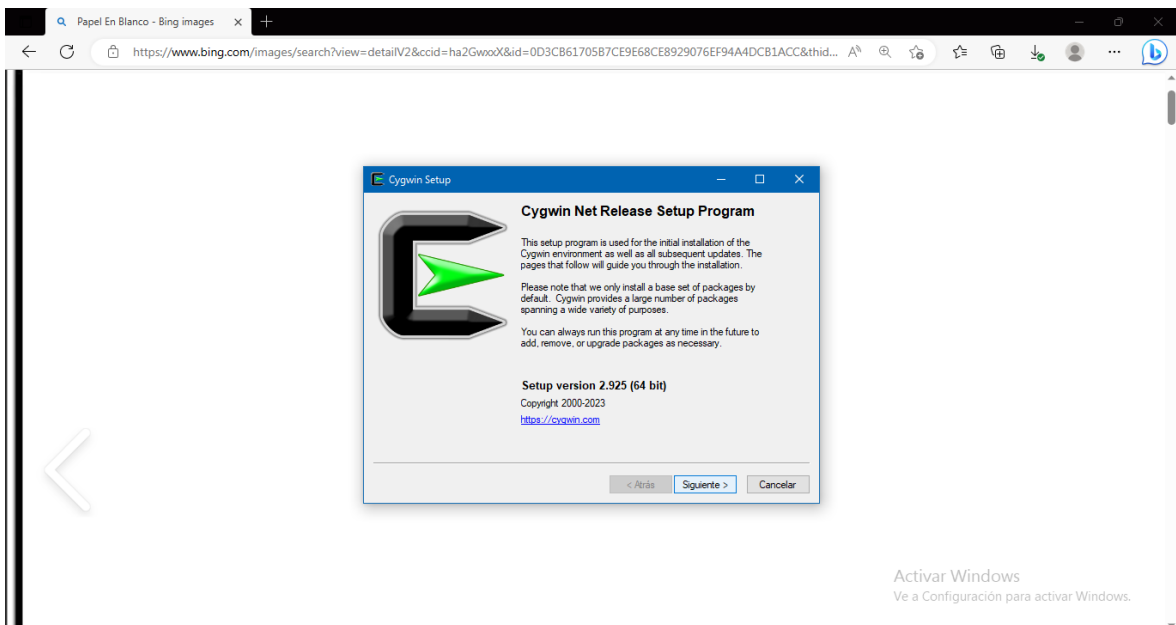
16. Ir al sitio [Cygwin Installation](https://cygwin.com/install.html). En el caso de Windows de 64 bits ejecute `setup-x86_64.exe` y para el caso de Windows de 32 bits ejecute `setup-x86_64.exe`



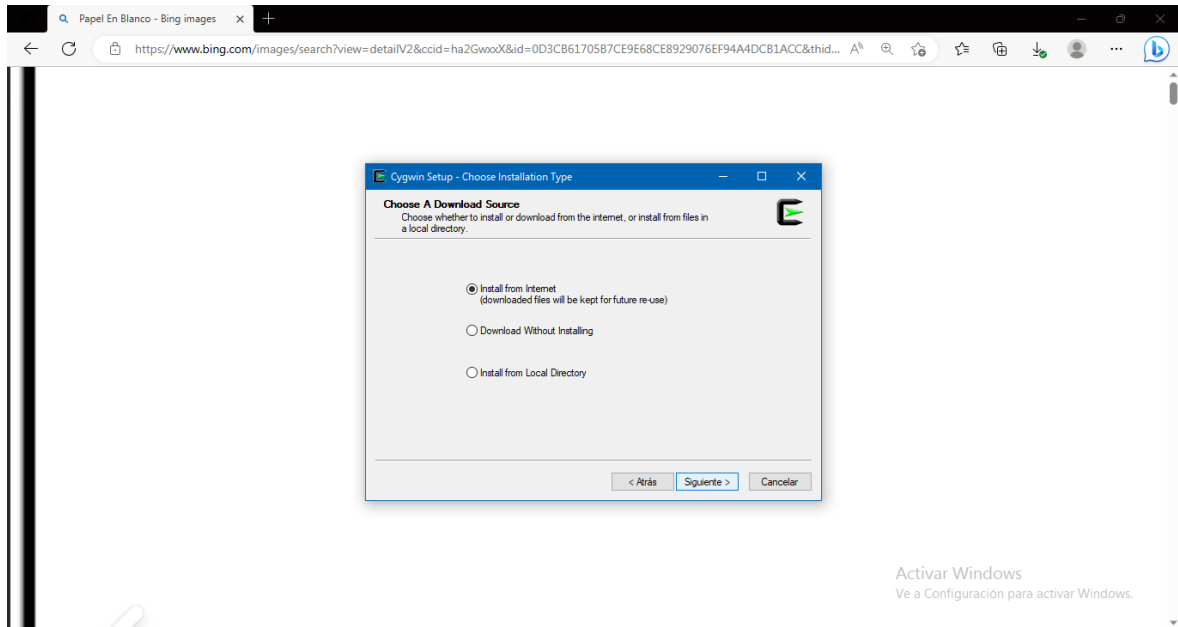
17. Dirijase a la ubicación de la descarga del archivo `.exe` y ejecutar como administrador.



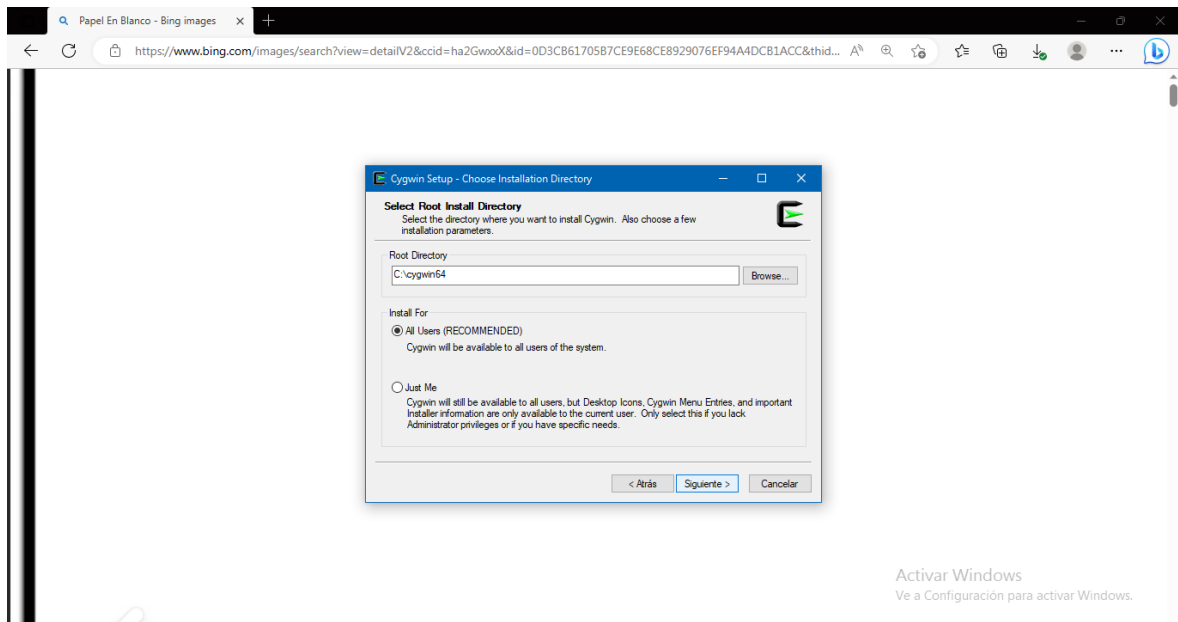
18. Clic en Siguiente.



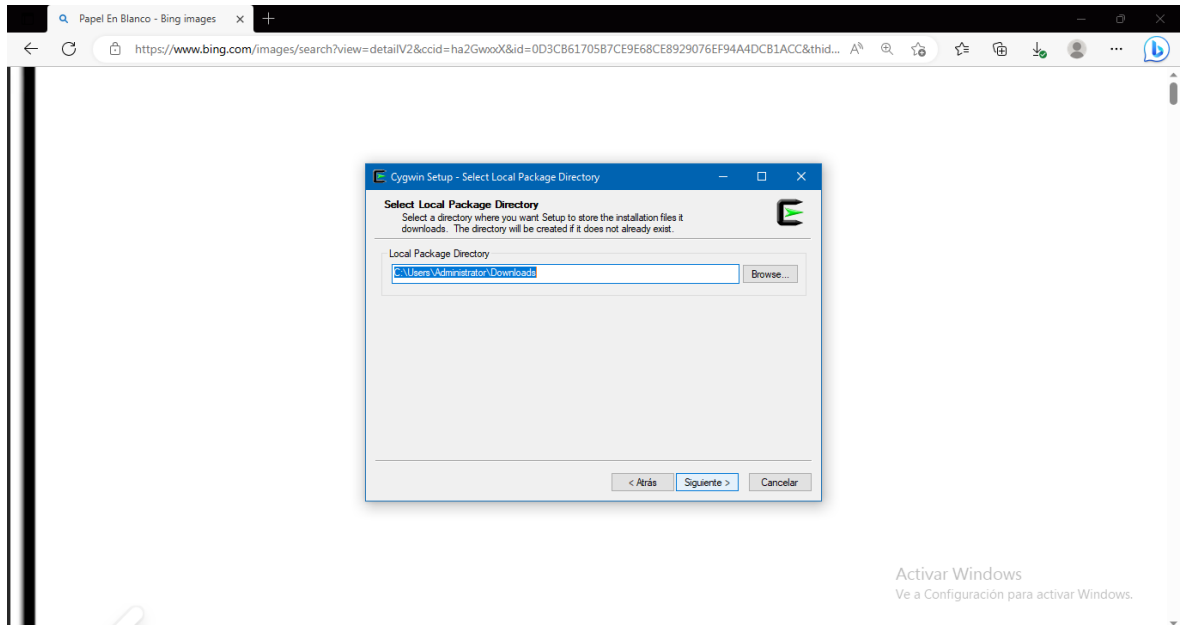
19. Seleccionar Install from Internet y Siguiente.



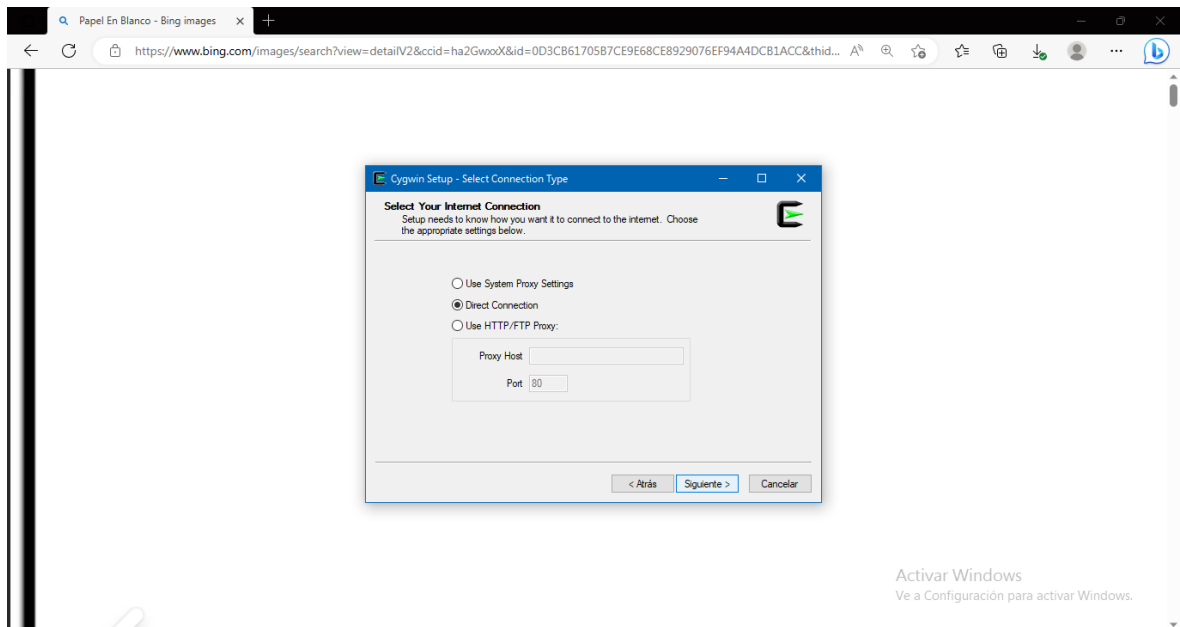
20. Dejar la ubicación de instalación por default en el disco local C y Siguiete.



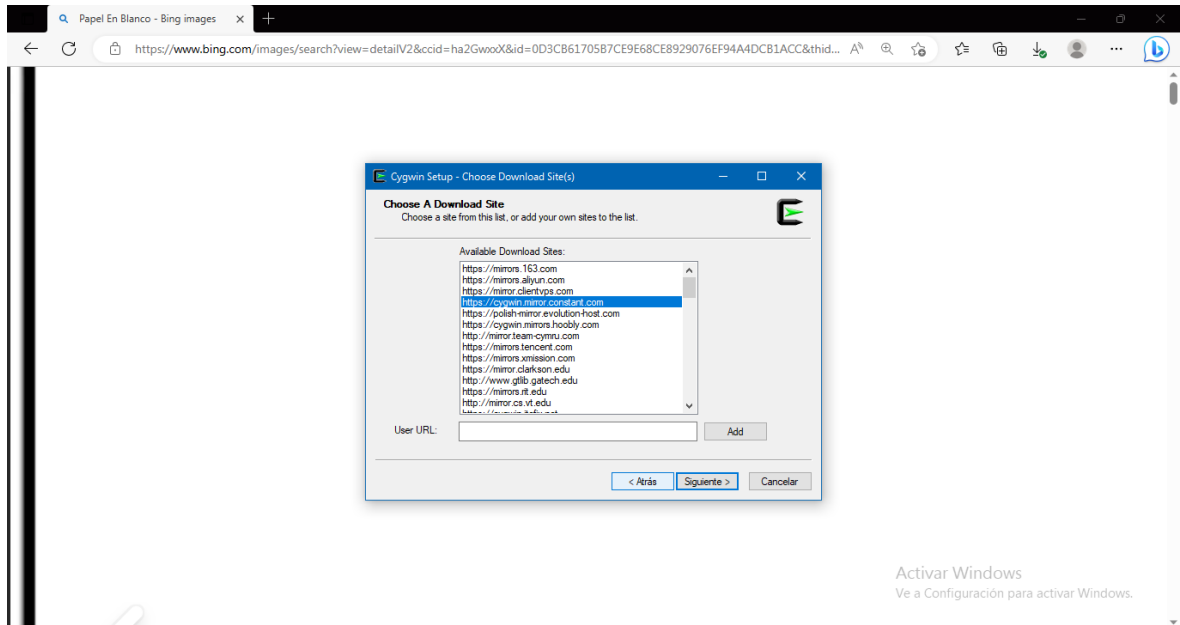
21. Dejar por default el directorio y Siguiete.



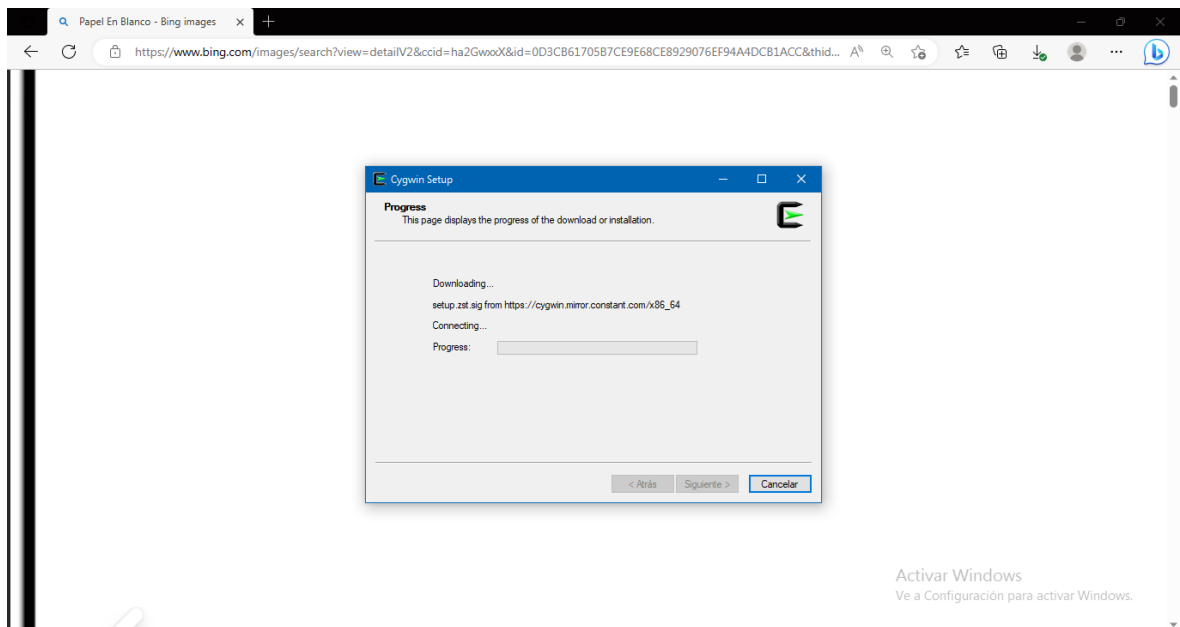
22. Elegir la opción Direct Connection y Siguiete.



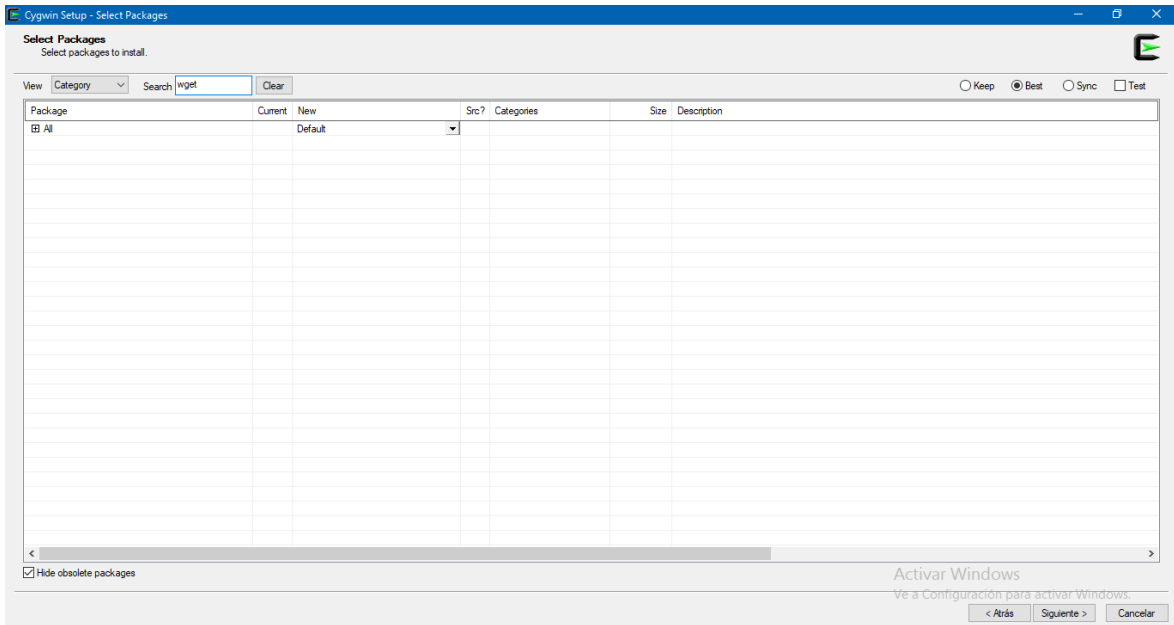
23. Elegir <http://cygwin.mirror.constant.com> y clic en Siguiete.



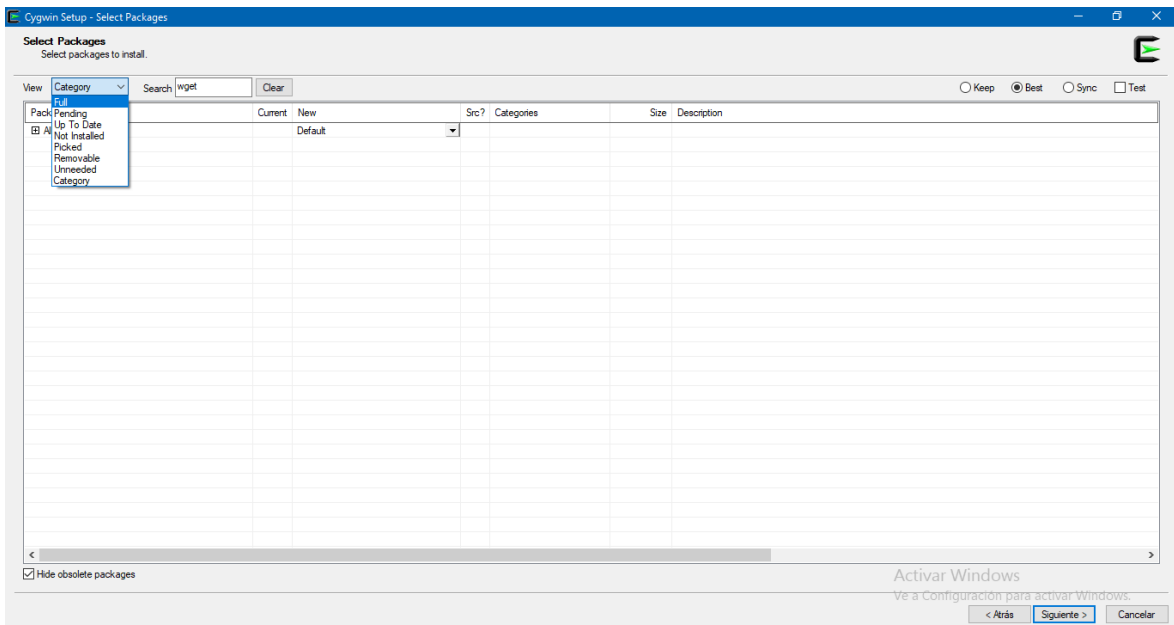
24. Esperar la descarga.



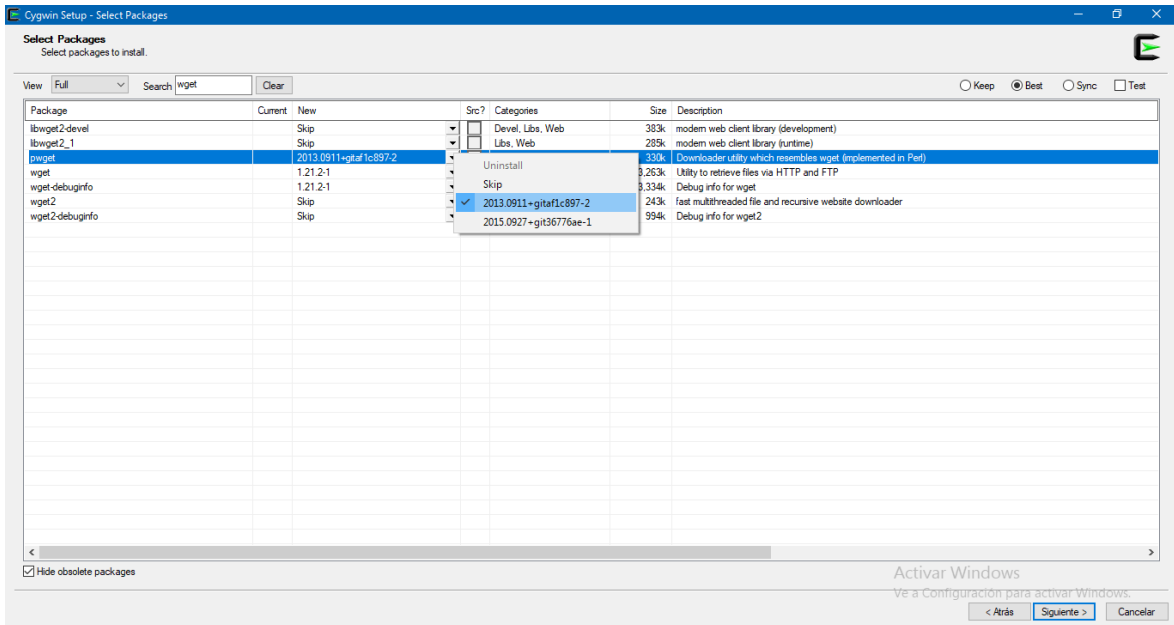
25. Al terminar la descarga, se abrirá un cuadro de dialogo. Escribir wget y buscar.



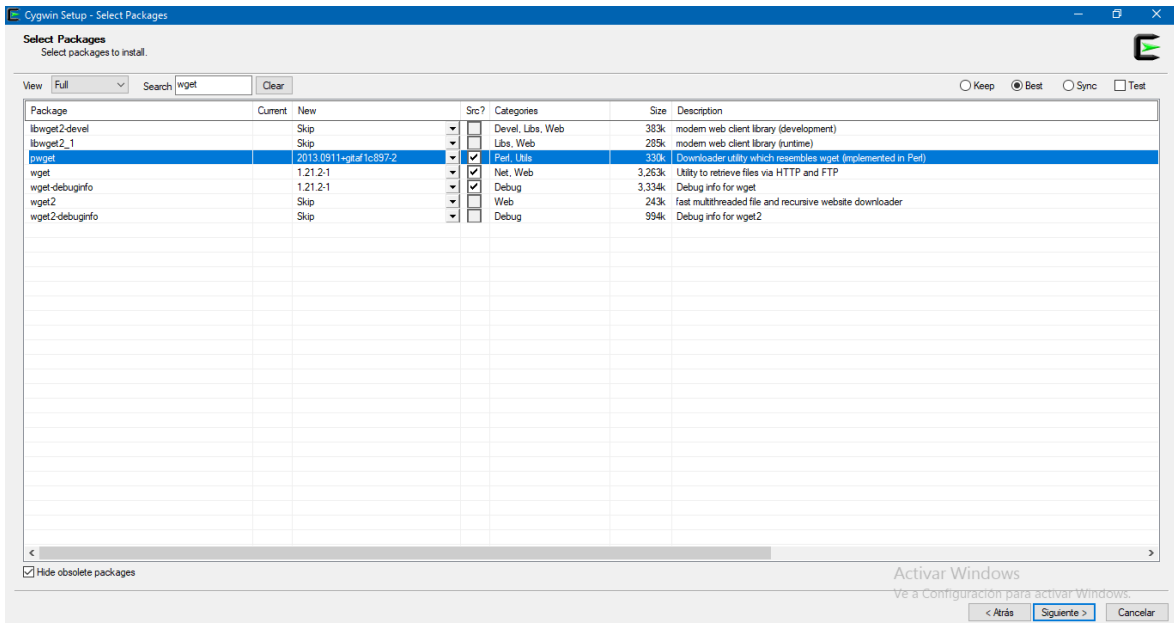
26. En View elegir la opción Full.



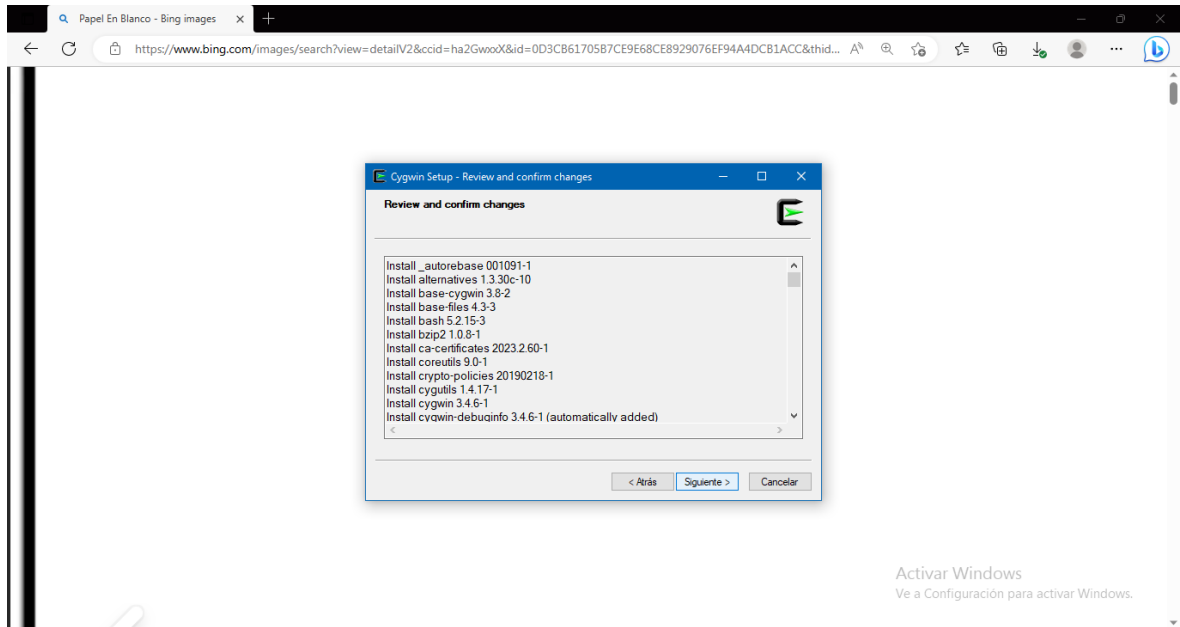
27. Se mostrara una lista con varios Items. Elegir pwget, wget, wget-debuginf para cambiar de Skip a la versión más actualizada en la columna New.



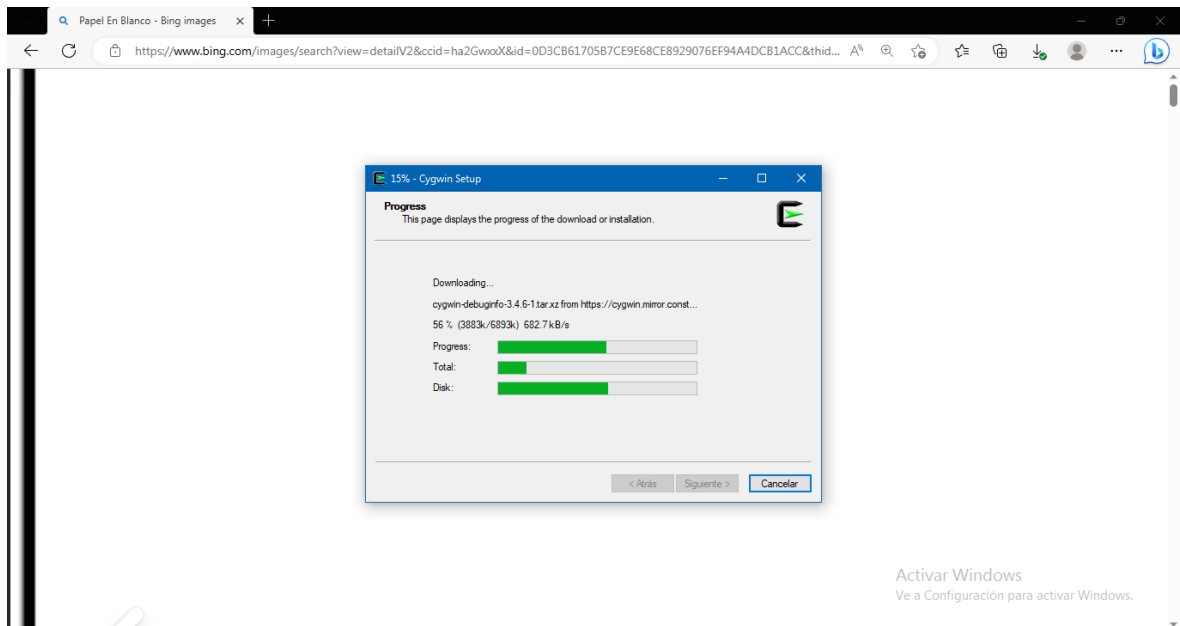
28. Marcar estos últimos y clic en Siguiente



29. Clic en Siguiente



30. Esperar la descarga.



DESCARGA DE LOS DATOS CON CYGWIN

31. Abrir el documento de texto con los enlaces del GES DISC.


```
Administrator@WIN-C1K7KK3PD98 ~
$ |
```

Activar Windows
Ve a Configuración para activar Windows.

34. Escribir `cd:/Ruta de la carpeta en la que está el archivo .dat` y enter.

```
/cygdrive/c/wgetdown
Administrator@WIN-C1K7KK3PD98 ~
$ cd C:/wgetdown
Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ |
```

Activar Windows
Ve a Configuración para activar Windows.

35. Revisar que la ruta sea correcta con el comando `pwd`.

```
Administrator@WIN-C1K7KK3PD98 ~
$ cd C:/wgetdown

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ pwd
/cygdrive/c/wgetdown

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ |
```

Activar Windows
Ve a Configuración para activar Windows.

36. Ejecutar: touch .netrc

```
Administrator@WIN-C1K7KK3PD98 ~
$ cd C:/wgetdown

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ pwd
/cygdrive/c/wgetdown

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ touch .netrc

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$
```

Activar Windows
Ve a Configuración para activar Windows.

37. Ejecutar: echo "machine urs.earthdata.nasa.gov login USERNAME password
PASSWORD" >>.netrc

```
Administrator@WIN-C1K7KK3PD98 ~
$ cd C:/wgetdown

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ pwd
/cygdrive/c/wgetdown

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ touch .netrc

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ echo "machine urs.earthdata.nasa.gov login Luis_marcelovargas password Luis1501">> .netrc

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$
```

Activar Windows
Ve a Configuración para activar Windows.

38. Ejecutar: chmod 0600 .netrc

```
Administrator@WIN-C1K7KK3PD98 ~
$ cd C:/wgetdown

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ pwd
/cygdrive/c/wgetdown

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ touch .netrc

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ echo "machine urs.earthdata.nasa.gov login Luis_marcelovargas password Luis1501">> .netrc

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ chmod 0600 .netrc

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ |
```

Activar Windows
Ve a Configuración para activar Windows.

39. Ejecutar: touch .urs_cookies


```

Administrator@WIN-C1K7KK3PD98 ~
$ cd C:/wgetdown

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ pwd
/cygdrive/c/wgetdown

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ touch .netrc

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ echo "machine urs.earthdata.nasa.gov login Luis_marcelovargas password Luis1501" >>.netrc

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ chmod 0600 .netrc

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ touch .urs_cookies

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$

```

40. Ejecutar: `wget --load-cookies .urs_cookies --save-cookies .urs_cookies --auth-no-challenge=on --keep-session-cookies --content-disposition --no-check-certificate --user=USERNAME --ask-password -i FILENAME.dat`

```

Administrator@WIN-C1K7KK3PD98 ~
$ cd C:/wgetdown

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ pwd
/cygdrive/c/wgetdown

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ touch .netrc

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ echo "machine urs.earthdata.nasa.gov login Luis_marcelovargas password Luis1501" >>.netrc

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ chmod 0600 .netrc

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ touch .urs_cookies

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ wget --load-cookies .urs_cookies --save-cookies .urs_cookies --auth-no-challenge=on --keep-session-cookies --content-disposition --no-check-certificate --user=Luis_marcelovargas --ask-passwo
rd -i datos.dat
wget: unknown option -- 1
Usage: wget [OPTION]... [URL]...

Try 'wget --help' for more options.

Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ wget --load-cookies .urs_cookies --save-cookies .urs_cookies --auth-no-challenge=on --keep-session-cookies --content-disposition --no-check-certificate --user=Luis_marcelovargas --ask-passwo
rd -i datos.dat
Password for user 'Luis_marcelovargas':

```

41. Escribir la contraseña del GES DISC.

```

C:\cygdrive/c/wgetdown
Administrator@WIN-C1K7KK3PD98 ~
$ cd C:/wgetdown
Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ pwd
/cygdrive/c/wgetdown
Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ touch .netrc
Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ echo "machine urs.earthdata.nasa.gov login Luis_marcelovargas password Luis1501" >>.netrc
Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ chmod 0600 .netrc
Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ touch .urs_cookies
Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ wget --load-cookies .urs_cookies --save-cookies .urs_cookies --auth-no-challenge-on --keep-session-cookies --content-disposition --no-check-certificate --user=Luis_marcelovargas --ask-pasw
rd -i datos.dat
wget's unknown option --i
Usage: wget [OPTION]... [URL]...
Try 'wget --help' for more options.
Administrator@WIN-C1K7KK3PD98 /cygdrive/c/wgetdown
$ wget --load-cookies .urs_cookies --save-cookies .urs_cookies --auth-no-challenge-on --keep-session-cookies --content-disposition --no-check-certificate --user=Luis_marcelovargas --ask-pasw
rd -i datos.dat
Password for user 'Luis_marcelovargas':
--2023-05-17 22:22:42-- https://docserver.gesdisc.eosdis.nasa.gov/public/project/GPM/IMERG_ATBD_V06.pdf
Resolving docserver.gesdisc.eosdis.nasa.gov (docserver.gesdisc.eosdis.nasa.gov)... 2001:4d0:241a:4041::32, 198.118.197.32
Connecting to docserver.gesdisc.eosdis.nasa.gov (docserver.gesdisc.eosdis.nasa.gov) [2001:4d0:241a:4041::32]:443...

```

42. Esperar la descarga.

```

C:\cygdrive/c/wgetdown
Connecting to docserver.gesdisc.eosdis.nasa.gov (docserver.gesdisc.eosdis.nasa.gov) [198.118.197.32]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2486295 (2.4M) [application/pdf]
Saving to: 'IMERG_ATBD_V06.pdf.1'

IMERG_ATBD_V06.pdf.1 100%[=====] 2.37M 587KB/s in 4.1s
2023-05-17 22:23:09 (587 KB/s) - 'IMERG_ATBD_V06.pdf.1' saved [2486295/2486295]

--2023-05-17 22:23:09-- https://gpm1.gesdisc.eosdis.nasa.gov/data/GPM_L3/doc/README.GPM.pdf
Resolving gpm1.gesdisc.eosdis.nasa.gov (gpm1.gesdisc.eosdis.nasa.gov)... 2001:4d0:241a:4041::50, 198.118.197.50
Connecting to gpm1.gesdisc.eosdis.nasa.gov (gpm1.gesdisc.eosdis.nasa.gov) [2001:4d0:241a:4041::50]:443... Failed: Connection timed out.
Connecting to gpm1.gesdisc.eosdis.nasa.gov (gpm1.gesdisc.eosdis.nasa.gov) [198.118.197.50]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 400392 (391K) [application/pdf]
Saving to: 'README.GPM.pdf.1'

README.GPM.pdf.1 100%[=====] 391.01K 262KB/s in 1.5s
2023-05-17 22:23:32 (262 KB/s) - 'README.GPM.pdf.1' saved [400392/400392]

--2023-05-17 22:23:32-- https://gpm1.gesdisc.eosdis.nasa.gov/daac-bin/OTF/HTTP_services.cgi?FILENAME=%2Fdata%2FGPM_L3%2FGPM_3IMERGHH_06%2F2000%2F15%2F3B-HHR.MS.MRG.3IMERG.20000601-5000000-E0
02959.0000.V06B.HDF5.SUB.nc40DATASET_VERSION=06SERVICEL34RS_GPM
Connecting to gpm1.gesdisc.eosdis.nasa.gov (gpm1.gesdisc.eosdis.nasa.gov) [198.118.197.50]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 30387 (30K) [application/octet-stream]
Saving to: '3B-HHR.MS.MRG.3IMERG.20000601-5000000-E002959.0000.V06B.HDF5.SUB.nc4.1'

3B-HHR.MS.MRG.3IMERG.20000601-5000000-E002959.0 100%[=====] 29.67K 138KB/s in 0.2s
2023-05-17 22:23:35 (138 KB/s) - '3B-HHR.MS.MRG.3IMERG.20000601-5000000-E002959.0000.V06B.HDF5.SUB.nc4.1' saved [30387/30387]

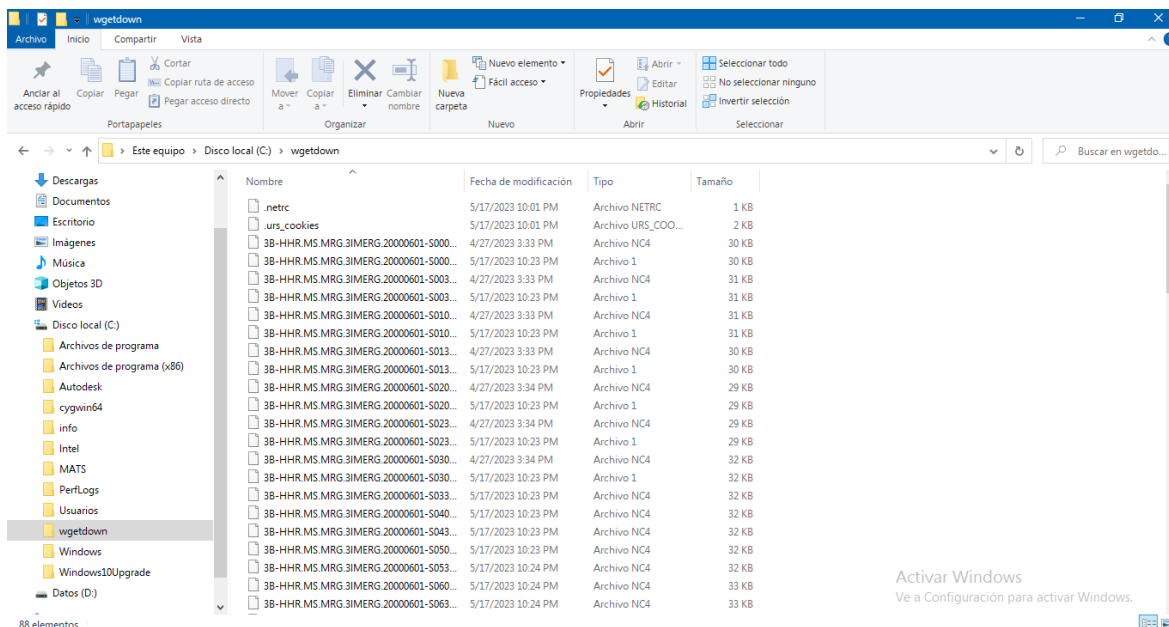
--2023-05-17 22:23:35-- https://gpm1.gesdisc.eosdis.nasa.gov/daac-bin/OTF/HTTP_services.cgi?FILENAME=%2Fdata%2FGPM_L3%2FGPM_3IMERGHH_06%2F2000%2F15%2F3B-HHR.MS.MRG.3IMERG.20000601-5003000-E0
05959.0030.V06B.HDF5.SUB.nc40DATASET_VERSION=06SERVICEL34RS_GPM
Connecting to gpm1.gesdisc.eosdis.nasa.gov (gpm1.gesdisc.eosdis.nasa.gov) [198.118.197.50]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 31230 (30K) [application/octet-stream]
Saving to: '3B-HHR.MS.MRG.3IMERG.20000601-5003000-E005959.0030.V06B.HDF5.SUB.nc4.1'

3B-HHR.MS.MRG.3IMERG.20000601-5003000-E005959.0 100%[=====] 30.50K 169KB/s in 0.2s
2023-05-17 22:23:37 (169 KB/s) - '3B-HHR.MS.MRG.3IMERG.20000601-5003000-E005959.0030.V06B.HDF5.SUB.nc4.1' saved [31230/31230]

--2023-05-17 22:23:37-- https://gpm1.gesdisc.eosdis.nasa.gov/daac-bin/OTF/HTTP_services.cgi?FILENAME=%2Fdata%2FGPM_L3%2FGPM_3IMERGHH_06%2F2000%2F15%2F3B-HHR.MS.MRG.3IMERG.20000601-5010000-E0
12959.0060.V06B.HDF5.SUB.nc40DATASET_VERSION=06SERVICEL34RS_GPM
Connecting to gpm1.gesdisc.eosdis.nasa.gov (gpm1.gesdisc.eosdis.nasa.gov) [198.118.197.50]:443... connected.
HTTP request sent, awaiting response...

```

43. Revisar en la carpeta que se estén descargando ahí los archivos.



Anexo 2. Script Abrir IMERG V06

```
#----- TIC IDF -----
#----- Abrir IMERG -----

#####

##1. ABRIR ARCHIVOS NC4
#####

#Remover las variables antiguas
rm(list=ls())

#Cargar las librerias
library(ncdf4)
library(sp)
library(raster)
library(rgdal)
library(zoo)
library(ggplot2)
library(scales)
library(lubridate)
library(dplyr)
library(hydroGOF)

#Seleccionamos el directorio de los archivos
```

```

setwd("D:/IMERGV06 FINAL 2000&2021 DATA")

#Hacemos una lista de los nombres de los archivos
f_names = list.files(pattern = ".nc4")

#Tabla que contenga las coordenadas geográficas de todas las estaciones
est_xy =
data.frame(c("M003","M006","M007","M025","M033","C02","C04","C05","C14","C15","P03",
", "P08","P09","P11","P12","P27","P28","P31","P32","P70"),
           c(-78.555,-79.461667,-75.403,-79.380278,-79.2034,-78.53747599,-
78.53128697,-78.4617,-78.58811972,-78.68532297,-78.52670301,-78.50994299,-
78.48767998,-78.52400403,-78.52726996,-78.53968498,-78.53797397,-78.57154195,-
78.59786293,-78.4973633),
           c(-0.365833,-1.074361,-0.9167,-0.0247222,-4.0375,-0.12974998,-
0.1754149,-0.18163888,-0.14942197,0.05982696,-0.1306779,-0.18091196,-0.17835395,-
0.16602191,-0.2062809,-0.20219093,-0.18224589,-0.11581599,-0.13640933,-
0.1616909))

colnames(est_xy) =c("nombre","lon","lat")

#Abrir nc4
pix0 =NA

for (i in 1:length(f_names)){
  #plot original IMERG
  ncfname = f_names[i]
  #ncin <- nc_open(ncfname)
  #print(ncin)
  #plot original
  r1 <- raster(ncfname)
  #r1
  #plot(r1)
  #transpose axis x-> y, x-> y
  rr1 = t(r1)
  #plot(rr1)
  #flip x and y directions
  rr1 = flip(rr1,direction="y")
  #plot(rr1)
  rr1 = flip(rr1,direction="x")
  #plot(rr1)
}

```

```

#rr1
#rr1@extent
#ncvar_get(ncin,"lon")
#ncvar_get(ncin,"lat")
#assign coordinate reference system (crs)
crs(rr1) <- "+init=epsg:4326 +proj=longlat +ellps=WGS84
+datum=WGS84 +no_defs +towgs84=0,0,0"
#extract one pixel value from raster
imerg0 =NA
for (j in 1:nrow(est_xy)){
  xy = est_xy[j,2:3] #(lon, lat)
  colnames(xy) = c("lon","lat")
  SpatialPoints(xy,proj4string=CRS("+init=epsg:4326 +proj=longlat +ellps=WGS84
+datum=WGS84 +no_defs +towgs84=0,0,0"))
  pixel1 = extract(rr1, SpatialPoints(xy), sp = T)
  #print(pixel1$layer)
  p1=pixel1$layer
  imerg0 = cbind.data.frame(imerg0,p1)
  pix1 = imerg0
}
pix0 = rbind.data.frame(pix0,pix1)
}
#Cambiar el nombre de las columnas
imerg_p = pix0[-1,-1]
colnames(imerg_p) =
c("M003IM","M006IM","M007IM","M025IM","M033IM","C02IM","C04IM","C05IM","C14IM","
C15IM","P03IM","P08IM","P09IM","P11IM","P12IM","P27IM","P28IM","P31IM","P32IM","P
70IM")
#####
##2. CREAR DATA FRAME IMERG V06
#####

#Crear el índice

```

```

start = as.POSIXct("2000-06-01 00:30:00", format="%Y-%m-%d %H:%M:%S")-5*60*60
#correct UTC a UTC-5

end = as.POSIXct("2021-09-04 23:30:00", format="%Y-%m-%d %H:%M:%S")-5*60*60
#correct UTC a UTC-5

index = seq(start,end,by="30 min")

dummy_z <- zoo(rep(NA,length(index)),index) # objeto zoo vacio

M003_z = zoo(imerg_p$M003,index)
M006_z = zoo(imerg_p$M006,index)
M007_z = zoo(imerg_p$M007,index)
M025_z = zoo(imerg_p$M025,index)
M033_z = zoo(imerg_p$M033,index)
C02_z = zoo(imerg_p$C02,index)
C04_z = zoo(imerg_p$C04,index)
C05_z = zoo(imerg_p$C05,index)
C14_z = zoo(imerg_p$C14,index)
C15_z = zoo(imerg_p$C15,index)
P03_z = zoo(imerg_p$P03,index)
P08_z = zoo(imerg_p$P08,index)
P09_z = zoo(imerg_p$P09,index)
P11_z = zoo(imerg_p$P11,index)
P12_z = zoo(imerg_p$P12,index)
P27_z = zoo(imerg_p$P27,index)
P28_z = zoo(imerg_p$P28,index)
P31_z = zoo(imerg_p$P31,index)
P32_z = zoo(imerg_p$P32,index)
P70_z = zoo(imerg_p$P70,index)

#merge objetos zoo

open_z <- merge(dummy_z,
M003_z,M006_z,M007_z,M025_z,M033_z,C02_z,C04_z,C05_z,C14_z,C15_z,P03_z,P08
_z,P09_z,P11_z,P12_z,P27_z,P28_z,P31_z,P32_z,P70_z)

open_df = as.data.frame(open_z)

open_df = open_df[,-1]

```

```

save(open_z,file="open_z77.RData")
save(open_df,file="open_df77.RData")
load("open_z77.RData")
load("open_df77.RData")
#Cambio de de unidades de Intensidad a Precipitacion
open_z11072023 = open_z*30*(1/60)
open_df11072023 = as.data.frame(open_z11072023 )

colnames(open_df11072023) =
c("M003IM","M006IM","M007IM","M025IM","M033IM","C02IM","C04IM","C05IM","C14IM","
C15IM","P03IM","P08IM","P09IM","P11IM","P12IM","P27IM","P28IM","P31IM","P32IM","P
70IM")

open_df11072023 = open_df11072023[,-1]
save(open_z11072023,file="open_z14072023.RData")
save(open_df11072023,file="open_df14072023.RData")
#load("open_z77.RData")
#load("open_z14072023.RData")
#####

##3. SERIES EN LOS DIFERENTES TIEMPOS
#####

#-----

#hacer índice cada 60 minutos

ind_1hr <- strptime("1970-01-01", "%Y-%m-%d") +
ceiling(as.numeric(index(open_z11072023))/(1*60*60))*(1*60*60)-5*60*60 #sum 1-5->5,
6-10 -> 10, 11-15->15, 16-20->20

zoo1hr_sum <- aggregate(open_z11072023,ind_1hr,function(x) round(sum(x,na.rm =
F),1))

#converted_index2 <- as.POSIXlt(index(zoo30min_sum), tz = "America/Bogota")-5*60*60

zoo1hr_open <- zoo1hr_sum

open_df1hr = data.frame(index(zoo1hr_open),zoo1hr_open)

open_df1hr11072023 = open_df1hr[,-2]

colnames(open_df1hr11072023) =c("time",
"M003IM","M006IM","M007IM","M025IM","M033IM","C02IM","C04IM","C05IM","C14IM","C
15IM","P03IM","P08IM","P09IM","P11IM","P12IM","P27IM","P28IM","P31IM","P32IM","P70
IM")

save(zoo1hr_open ,file="open_z1hr14072023.RData")

```

```

save(open_df1hr11072023,file="open_df1hr14072023.RData")

#load("open_df1hr14072023.RData")

#-----

#hacer índice cada 2 horas

ind_2hr <- strptime("1970-01-01", "%Y-%m-%d") +
ceiling(as.numeric(index(open_z11072023))/(2*60*60))*(2*60*60)-5*60*60 #sum 1-5->5,
6-10 -> 10, 11-15->15, 16-20->20

zoo2hr_sum <- aggregate(open_z11072023,ind_2hr,function(x) round(sum(x,na.rm =
F),1))

#converted_index2 <- as.POSIXlt(index(zoo30min_sum), tz = "America/Bogota")-5*60*60

zoo2hr_open <- zoo2hr_sum

open_df2hr11072023 = data.frame(index(zoo2hr_open),zoo2hr_open)

open_df2hr11072023 = open_df2hr11072023[,-2]

colnames(open_df2hr11072023) =c("time",
"M003IM","M006IM","M007IM","M025IM","M033IM","C02IM","C04IM","C05IM","C14IM","C
15IM","P03IM","P08IM","P09IM","P11IM","P12IM","P27IM","P28IM","P31IM","P32IM","P70
IM")

save(zoo2hr_open ,file="open_z2hr14072023.RData")

save(open_df2hr11072023,file="open_df2hr14072023.RData")

#load("open_df2hr14072023.RData")

#-----

#hacer índice cada 6 horas

ind_6hr <- strptime("1970-01-01", "%Y-%m-%d") +
ceiling(as.numeric(index(open_z11072023))/(6*60*60))*(6*60*60)-5*60*60 #sum 1-5->5,
6-10 -> 10, 11-15->15, 16-20->20

zoo6hr_sum <- aggregate(open_z11072023,ind_6hr,function(x) round(sum(x,na.rm =
F),1))

#converted_index2 <- as.POSIXlt(index(zoo30min_sum), tz = "America/Bogota")-5*60*60

zoo6hr_open <- zoo6hr_sum

open_df6hr11072023 = data.frame(index(zoo6hr_open),zoo6hr_open)

open_df6hr11072023 = open_df6hr11072023[,-2]

colnames(open_df6hr11072023) =c("time",
"M003IM","M006IM","M007IM","M025IM","M033IM","C02IM","C04IM","C05IM","C14IM","C
15IM","P03IM","P08IM","P09IM","P11IM","P12IM","P27IM","P28IM","P31IM","P32IM","P70
IM")

```



```

save(zoo6hr_open ,file="open_z6hr14072023.RData")
save(open_df6hr11072023,file="open_df6hr14072023.RData")
#load("open_df6hr14072023.RData")
#-----
#hacer índice cada 12 horas
ind_12hr <- strptime("1970-01-01", "%Y-%m-%d") +
ceiling(as.numeric(index(open_z11072023))/(12*60*60))*(12*60*60)-5*60*60 #sum 1-5-
>5, 6-10 -> 10, 11-15->15, 16-20->20
zoo12hr_sum <- aggregate(open_z11072023,ind_12hr,function(x) round(sum(x,na.rm =
F),1))
#converted_index2 <- as.POSIXlt(index(zoo30min_sum), tz = "America/Bogota")-5*60*60
zoo12hr_open <- zoo12hr_sum
open_df12hr11072023 = data.frame(index(zoo12hr_open),zoo12hr_open)
open_df12hr11072023 = open_df12hr11072023[,-2]
colnames(open_df12hr11072023) =c("time",
"M003IM","M006IM","M007IM","M025IM","M033IM","C02IM","C04IM","C05IM","C14IM","C
15IM","P03IM","P08IM","P09IM","P11IM","P12IM","P27IM","P28IM","P31IM","P32IM","P70
IM")
save(zoo12hr_open ,file="open_z12hr14072023.RData")
save(open_df12hr11072023,file="open_df12hr11072023.RData")
#load("open_df12hr14072023.RData")
#-----
#hacer índice cada 24 horas
ind_24hr <- strptime("1970-01-01", "%Y-%m-%d") +
ceiling(as.numeric(index(open_z11072023))/(24*60*60))*(24*60*60)-5*60*60 #sum 1-5-
>5, 6-10 -> 10, 11-15->15, 16-20->20
zoo24hr_sum <- aggregate(open_z11072023,ind_24hr,function(x) round(sum(x,na.rm =
F),1))
#converted_index2 <- as.POSIXlt(index(zoo30min_sum), tz = "America/Bogota")-5*60*60
zoo24hr_open <- zoo24hr_sum
open_df24hr11072023 = data.frame(index(zoo24hr_open),zoo24hr_open)
open_df24hr11072023 = open_df24hr11072023[,-2]
colnames(open_df24hr11072023) =c("time",
"M003IM","M006IM","M007IM","M025IM","M033IM","C02IM","C04IM","C05IM","C14IM","C
15IM","P03IM","P08IM","P09IM","P11IM","P12IM","P27IM","P28IM","P31IM","P32IM","P70
IM")

```

```

save(zoo24hr_open ,file="open_z24hr14072023.RData")
save(open_df24hr11072023,file="open_df24hr11072023.RData")
#load("open_df24hr14072023.RData")
#-----

Anexo 3. Script agregación para las series temporales a diferentes duraciones

#----- TIC SPP IDFs -----
#----- AGREGACION OBSERVADA -----

#Aragación datos lluvia

library(xts)

library(zoo)

library(stringr)

library(ggplot2)

library(hydroGOF)

library(lubridate)

library(dplyr)

#####

##1. Abrir archivos EPMAPS

#####

#Sys.setlocale("LC_TIME", "English")

#borrar variables anteriores

rm(list=ls())

#guardar los nombres de las estaciones en un vector, deben estar los archivos .csv en la
carpeta del proyecto

epmaps<-list.files(pattern = "EPMAPS")

#modificar el nombre, eliminar todos los numeros del nombre y el guion medio "-"

x<-substr(epmaps,-2,nchar(epmaps)-38)

est_epmaps<-gsub('-', '_',x)

```

```

est_epmaps<-gsub(' ','_',est_epmaps)
est_epmaps<-gsub('Ñ','N',est_epmaps)
#poner la letra m delante del nombre del objeto y abrir los csv
for (i in 1: length(epmaps)){
  data = read.csv(epmaps[i],header=T)
  assign(paste('m',est_epmaps[i],sep='_'),data)
}

#convertir a as.POSIXct la fecha
#pasar a elemento de clase zoo
date_max<-as.character()
date_min<-as.character()
for (i in 1:length(ls(pattern='^m'))){
  a<-get(ls(pattern='^m')[i])
  a[,1]<-as.POSIXct(a[,1],format='%Y/%m/%d %H:%M:%S')
  #a[,1]<-round(a[,1], unit="mins")
  #borrar duplicado
  borrar <- which(duplicated(a[,1]))
  if (length(borrar) > 0) {
    a <- a[-borrar,]
  }
  #crear zoo
  date_min[i]<-as.character(a[1,1])
  date_max[i]<-as.character(a[nrow(a),1])
  a_z<-zoo(a[,2],a[,1])
  assign(paste('z',est_epmaps[i],sep='_'),a_z)
}

```

```

}

#####

##2. Crear data frame EPMAPS

#####

#start<-as.POSIXct("2000/05/02 09:45:00",format="%Y/%m/%d %H:%M:%S")
start = as.POSIXct(min(date_min))

#end<-as.POSIXct("2022/12/05 09:20:00",format="%Y/%m/%d %H:%M:%S")
end = as.POSIXct(max(date_max))

#Contar vacioss

#-----

#EPMAPS C02 RUMIHURCO

gapsc02_inicio = which(!is.na(z_EPMAPS_C02_RUMIHURCO))[1]

gapsc02_fin = which(!is.na(z_EPMAPS_C02_RUMIHURCO))[length(
which(!is.na(z_EPMAPS_C02_RUMIHURCO)))]

totalc02 = length(z_EPMAPS_C02_RUMIHURCO[gapsc02_inicio:gapsc02_fin])

gapsc02 =
length(which(is.na(z_EPMAPS_C02_RUMIHURCO)[gapsc02_inicio:gapsc02_fin]))

Pgapsc02 = gapsc02/totalc02*100

#-----

#z_EPMAPS_C04_RUMIPAMBA

gapsc04_inicio = which(!is.na(z_EPMAPS_C04_RUMIPAMBA))[1]

gapsc04_fin = which(!is.na(z_EPMAPS_C04_RUMIPAMBA))[length(
which(!is.na(z_EPMAPS_C04_RUMIPAMBA)))]

totalc04 = length(z_EPMAPS_C04_RUMIPAMBA[gapsc04_inicio:gapsc04_fin])

gapsc04 =
length(which(is.na(z_EPMAPS_C04_RUMIPAMBA)[gapsc04_inicio:gapsc04_fin]))

Pgapsc04 = gapsc04/totalc04*100

```

```

#-----
#z_EPMAPS_C05_BELLAVISTA

gapsc05_inicio = which(!is.na(z_EPMAPS_C05_BELLAVISTA))[1]

gapsc05_fin = which(!is.na(z_EPMAPS_C05_BELLAVISTA))[length(
which(!is.na(z_EPMAPS_C05_BELLAVISTA)))]

totalc05 = length(z_EPMAPS_C05_BELLAVISTA[gapsc05_inicio:gapsc05_fin])

gapsc05 =
length(which(is.na(z_EPMAPS_C05_BELLAVISTA)[gapsc05_inicio:gapsc05_fin]))

Pgapsc05 = gapsc05/totalc05*100

#-----
#z_EPMAPS_C14_MINDO_CAPTACION

gapsc14_inicio = which(!is.na(z_EPMAPS_C14_MINDO_CAPTACION))[1]

gapsc14_fin = which(!is.na(z_EPMAPS_C14_MINDO_CAPTACION))[length(
which(!is.na(z_EPMAPS_C14_MINDO_CAPTACION)))]

totalc14 = length(z_EPMAPS_C14_MINDO_CAPTACION[gapsc14_inicio:gapsc14_fin])

gapsc14 =
length(which(is.na(z_EPMAPS_C14_MINDO_CAPTACION)[gapsc14_inicio:gapsc14_fin])
)

Pgapsc14 = gapsc14/totalc14*100

#-----
#z_EPMAPS_C15_NANEGALITO

gapsc15_inicio = which(!is.na(z_EPMAPS_C15_NANEGALITO))[1]

gapsc15_fin = which(!is.na(z_EPMAPS_C15_NANEGALITO))[length(
which(!is.na(z_EPMAPS_C15_NANEGALITO)))]

totalc15 = length(z_EPMAPS_C15_NANEGALITO[gapsc15_inicio:gapsc15_fin])

gapsc15 =
length(which(is.na(z_EPMAPS_C15_NANEGALITO)[gapsc15_inicio:gapsc15_fin]))

Pgapsc15 = gapsc15/totalc15*100

```

```

#-----
#z_EPMAPS_P03_RUMIHURCO_MACHANGARA

gasp03_inicio = which(!is.na(z_EPMAPS_P03_RUMIHURCO_MACHANGARA))[1]

gasp03_fin = which(!is.na(z_EPMAPS_P03_RUMIHURCO_MACHANGARA))[length(
which(!is.na(z_EPMAPS_P03_RUMIHURCO_MACHANGARA)))]

totalp03 =
length(z_EPMAPS_P03_RUMIHURCO_MACHANGARA[gasp03_inicio:gasp03_fin])

gasp03 =
length(which(is.na(z_EPMAPS_P03_RUMIHURCO_MACHANGARA)[gasp03_inicio:gasp03_fin]))

Pgasp03 = gasp03/totalp03*100

```

```

#-----
#z_EPMAPS_P08_RUMIPAMBA_BODEGAS

gasp08_inicio = which(!is.na(z_EPMAPS_P08_RUMIPAMBA_BODEGAS))[1]

gasp08_fin = which(!is.na(z_EPMAPS_P08_RUMIPAMBA_BODEGAS))[length(
which(!is.na(z_EPMAPS_P08_RUMIPAMBA_BODEGAS)))]

totalp08 =
length(z_EPMAPS_P08_RUMIPAMBA_BODEGAS[gasp08_inicio:gasp08_fin])

gasp08 =
length(which(is.na(z_EPMAPS_P08_RUMIPAMBA_BODEGAS)[gasp08_inicio:gasp08_fin]))

Pgasp08 = gasp08/totalp08*100

```

```

#-----
#z_EPMAPS_P09_INAQUITO_INAMHI

gasp09_inicio = which(!is.na(z_EPMAPS_P09_INAQUITO_INAMHI))[1]

gasp09_fin = which(!is.na(z_EPMAPS_P09_INAQUITO_INAMHI))[length(
which(!is.na(z_EPMAPS_P09_INAQUITO_INAMHI)))]

totalp09 = length(z_EPMAPS_P09_INAQUITO_INAMHI[gasp09_inicio:gasp09_fin])

```

```

gasp09 =
length(which(is.na(z_EPMAPS_P09_INAQUITO_INAMHI)[gasp09_inicio:gasp09_fin]))

Pgasp09 = gasp09/totalp09*100

#-----

#z_EPMAPS_P11_ANTENAS

gasp11_inicio = which(!is.na(z_EPMAPS_P11_ANTENAS))[1]

gasp11_fin = which(!is.na(z_EPMAPS_P11_ANTENAS))[length(
which(!is.na(z_EPMAPS_P11_ANTENAS)))]

totalp11 = length(z_EPMAPS_P11_ANTENAS[gasp11_inicio:gasp11_fin])

gasp11 =
length(which(is.na(z_EPMAPS_P11_ANTENAS)[gasp11_inicio:gasp11_fin]))

Pgasp11 = gasp11/totalp11*100

#-----

#z_EPMAPS_P12_TOCTIUCO_Pr

gasp12_inicio = which(!is.na(z_EPMAPS_P12_TOCTIUCO_Pr))[1]

gasp12_fin = which(!is.na(z_EPMAPS_P12_TOCTIUCO_Pr))[length(
which(!is.na(z_EPMAPS_P12_TOCTIUCO_Pr)))]

totalp12 = length(z_EPMAPS_P12_TOCTIUCO_Pr[gasp12_inicio:gasp12_fin])

gasp12 =
length(which(is.na(z_EPMAPS_P12_TOCTIUCO_Pr)[gasp12_inicio:gasp12_fin]))

Pgasp12 = gasp12/totalp12*100

#-----

#z_EPMAPS_P27_SAN_FRANCISCO

gasp27_inicio = which(!is.na(z_EPMAPS_P27_SAN_FRANCISCO))[1]

gasp27_fin = which(!is.na(z_EPMAPS_P27_SAN_FRANCISCO))[length(
which(!is.na(z_EPMAPS_P27_SAN_FRANCISCO)))]

totalp27 = length(z_EPMAPS_P27_SAN_FRANCISCO[gasp27_inicio:gasp27_fin])

```

```
gapsp27 =  
length(which(is.na(z_EPMAPS_P27_SAN_FRANCISCO)[gapsp27_inicio:gapsp27_fin]))
```

```
Pgapsp27 = gapsp27/totalp27*100
```

```
#-----
```

```
#z_EPMAPS_P28_CRUZ_LOMA
```

```
gapsp28_inicio = which(!is.na(z_EPMAPS_P08_RUMIPAMBA_BODEGAS))[1]
```

```
gapsp28_fin = which(!is.na(z_EPMAPS_P08_RUMIPAMBA_BODEGAS))[length(  
which(!is.na(z_EPMAPS_P08_RUMIPAMBA_BODEGAS)))]
```

```
totalp28 =
```

```
length(z_EPMAPS_P08_RUMIPAMBA_BODEGAS[gapsp28_inicio:gapsp28_fin])
```

```
gapsp28 =
```

```
length(which(is.na(z_EPMAPS_P28_CRUZ_LOMA)[gapsp28_inicio:gapsp28_fin]))
```

```
Pgapsp28 = gapsp28/totalp28*100
```

```
#-----
```

```
#z_EPMAPS_P31_PICHAN
```

```
gapsp31_inicio = which(!is.na(z_EPMAPS_P31_PICHAN))[1]
```

```
gapsp31_fin = which(!is.na(z_EPMAPS_P31_PICHAN))[length(  
which(!is.na(z_EPMAPS_P31_PICHAN)))]
```

```
totalp31 = length(z_EPMAPS_P31_PICHAN[gapsp31_inicio:gapsp31_fin])
```

```
gapsp31 = length(which(is.na(z_EPMAPS_P31_PICHAN)[gapsp31_inicio:gapsp31_fin]))
```

```
Pgapsp31 = gapsp31/totalp31*100
```

```
#-----
```

```
#z_EPMAPS_P32_MINDO_BAJO))
```

```
gapsp32_inicio = which(!is.na(z_EPMAPS_P32_MINDO_BAJO))[1]
```

```
gapsp32_fin = which(!is.na(z_EPMAPS_P32_MINDO_BAJO))[length(  
which(!is.na(z_EPMAPS_P32_MINDO_BAJO)))]
```

```
totalp32 = length(z_EPMAPS_P32_MINDO_BAJO[gapsp32_inicio:gapsp32_fin])
```



```

gasp32 =
length(which(is.na(z_EPMAPS_P32_MINDO_BAJO)[gasp32_inicio:gasp32_fin]))

Pgasp32 = gasp32/totalp32*100

#-----

#z_EPMAPS_P70_CC_EL_BOSQUE

gasp70_inicio = which(!is.na(z_EPMAPS_P70_CC_EL_BOSQUE))[1]

gasp70_fin = which(!is.na(z_EPMAPS_P70_CC_EL_BOSQUE))[length(
which(!is.na(z_EPMAPS_P70_CC_EL_BOSQUE)))]

totalp70 = length(z_EPMAPS_P70_CC_EL_BOSQUE[gasp70_inicio:gasp70_fin])

gasp70 =
length(which(is.na(z_EPMAPS_P70_CC_EL_BOSQUE)[gasp70_inicio:gasp70_fin]))

Pgasp70 = gasp70/totalp70*100

#-----

#objeto zoo vacío

index<-seq(start,end,by="5 min")

dummy_z <- zoo(rep(NA,length(index)),index) # objeto zoo vacío

#merge objetos zoo

epmaps_z11072023 <- merge(dummy_z,
z_EPMAPS_C02_RUMIHURCO,z_EPMAPS_C04_RUMIPAMBA,z_EPMAPS_C05_BELL
AVISTA,z_EPMAPS_C14_MINDO_CAPTACION,z_EPMAPS_C15_NANEGALITO,z_EP
MAPS_P03_RUMIHURCO_MACHANGARA,z_EPMAPS_P08_RUMIPAMBA_BODEGAS,
z_EPMAPS_P09_INAQUITO_INAMHI,z_EPMAPS_P11_ANTENAS,z_EPMAPS_P12_TO
CTIUOCO_Pr,z_EPMAPS_P27_SAN_FRANCISCO,z_EPMAPS_P28_CRUZ_LOMA,z_EP
MAPS_P31_PICHAN,z_EPMAPS_P32_MINDO_BAJO,z_EPMAPS_P70_CC_EL_BOSQ
UE)

epmaps_df11072023 = as.data.frame(epmaps_z)

save(epmaps_z11072023, file="EPMAPS_z15072023.RData")

save(epmaps_df11072023, file="EPMAPS_df15072023.RData")

```

```

#load("EPMAPS_z15072023..RData")

#load("EPMAPS_df15072023.RData")

#####

#3. Agregación EPMAPS

#####

#Hacer indice cada 30min

ind_30min <- strptime("1970-01-01", "%Y-%m-%d") +
ceiling(as.numeric(index(epmaps_z11072023))/(30*60))*(30*60)-5*60*60 #sum 1-5->5, 6-
10 -> 10, 11-15->15, 16-20->20

zoo30min_sum <- aggregate(epmaps_z11072023,ind_30min,function(x)
round(sum(x,na.rm = F),1))

converted_index2 <- as.POSIXlt(index(zoo30min_sum), tz = "America/Bogota")-5*60*60

zoo30min_epmaps <- zoo30min_sum

epmaps_df30min = data.frame(index(zoo30min_epmaps),zoo30min_epmaps)

epmaps_df30min11072023 = epmaps_df30min[,-2]

colnames(epmaps_df30min11072023) =c("time",
"C02EPM","C04EPM","C05EPM","C14EPM","C15EPM","P03EPM","P08EPM","P09EPM",
"P11EPM","P12EPM","P27EPM","P28EPM","P31EPM","P32EPM","P70EPM"

save(epmaps_df30min11072023, file="EPMAPSdf30min15072023.RData")

save(zoo30min_epmaps, file="EPMAPSz30min15072023.RData")

#load("EPMAPSz30min15072023.RData")

#load("EPMAPSdf30min15072023.RData")

#-----

#Hacer indice cada hora

ind_1hr <- strptime("1970-01-01", "%Y-%m-%d") +
ceiling(as.numeric(index(epmaps_z11072023))/(1*60*60))*(1*60*60)-5*60*60 #sum 1-5-
>5, 6-10 -> 10, 11-15->15, 16-20->20

```

```

zoo1hr_sum <- aggregate(epmaps_z11072023,ind_1hr,function(x) round(sum(x,na.rm =
F),1))

converted_index2 <- as.POSIXlt(index(zoo1hr_sum), tz = "America/Bogota")-5*60*60

zoo1hr_epmaps <- zoo1hr_sum

epmaps_df1hr = data.frame(index(zoo1hr_epmaps),zoo1hr_epmaps)

epmaps_df1hr11072023 = epmaps_df1hr[,-2]

colnames(epmaps_df1hr11072023) =c("time",
"C02EPM","C04EPM","C05EPM","C14EPM","C15EPM","P03EPM","P08EPM","P09EPM",
"P11EPM","P12EPM","P27EPM","P28EPM","P31EPM","P32EPM","P70EPM")

save(epmaps_df1hr11072023, file="EPMAPSdf1hr15072023.RData")

save(zoo1hr_epmaps, file="EPMAPSz1hr15072023.RData")

#load("EPMAPSdf1hr15072023.RData")

#load("EPMAPSz1hr15072023.RData")

#-----

#hacer indice cada 2 horas

ind_2hr <- strptime("1970-01-01", "%Y-%m-%d") +
ceiling(as.numeric(index(epmaps_z11072023))/(2*60*60))*(2*60*60)-5*60*60 #sum 1-5-
>5, 6-10 -> 10, 11-15->15, 16-20->20

zoo2hr_sum <- aggregate(epmaps_z11072023,ind_2hr,function(x) round(sum(x,na.rm =
F),1))

converted_index2 <- as.POSIXlt(index(zoo2hr_sum), tz = "America/Bogota")-5*60*60

zoo2hr_epmaps <- zoo2hr_sum

epmaps_df2hr = data.frame(index(zoo2hr_epmaps),zoo2hr_epmaps)

epmaps_df2hr11072023 = epmaps_df2hr[,-2]

colnames(epmaps_df2hr11072023) =c("time",
"C02EPM","C04EPM","C05EPM","C14EPM","C15EPM","P03EPM","P08EPM","P09EPM",
"P11EPM","P12EPM","P27EPM","P28EPM","P31EPM","P32EPM","P70EPM")

save(epmaps_df2hr11072023, file="EPMAPSdf2hr15072023.RData")

```

```

save(zoo2hr_epmaps, file="EPMAPSz2hr15072023.RData")

#load("EPMAPSdf2hr15072023.RData")

#load("EPMAPSz2hr15072023.RData")

#-----

#hacer indice cada 6 horas

ind_6hr <- strptime("1970-01-01", "%Y-%m-%d") +
ceiling(as.numeric(index(epmaps_z11072023))/(6*60*60))*(6*60*60)-5*60*60 #sum 1-5-
>5, 6-10 -> 10, 11-15->15, 16-20->20

zoo6hr_sum <- aggregate(epmaps_z11072023,ind_6hr,function(x) round(sum(x,na.rm =
F),1))

converted_index2 <- as.POSIXlt(index(zoo6hr_sum), tz = "America/Bogota")-5*60*60

zoo6hr_epmaps <- zoo6hr_sum

epmaps_df6hr = data.frame(index(zoo6hr_epmaps),zoo6hr_epmaps)

epmaps_df6hr11072023 = epmaps_df6hr[,-2]

colnames(epmaps_df6hr11072023) =c("time",
"C02EPM","C04EPM","C05EPM","C14EPM","C15EPM","P03EPM","P08EPM","P09EPM",
"P11EPM","P12EPM","P27EPM","P28EPM","P31EPM","P32EPM","P70EPM")

save(epmaps_df6hr11072023, file="EPMAPSdf6hr15072023.RData")

save(zoo6hr_epmaps, file="EPMAPSz6hr15072023.RData")

#load("EPMAPSdf6hr15072023.RData")

#load("EPMAPSz6hr15072023.RData")

#-----

#hacer indice cada 12 horas

ind_12hr <- strptime("1970-01-01", "%Y-%m-%d") +
ceiling(as.numeric(index(epmaps_z11072023))/(12*60*60))*(12*60*60)-5*60*60 #sum 1-
5->5, 6-10 -> 10, 11-15->15, 16-20->20

zoo12hr_sum <- aggregate(epmaps_z11072023,ind_12hr,function(x) round(sum(x,na.rm
= F),1))

```

```

converted_index2 <- as.POSIXlt(index(zoo12hr_sum), tz = "America/Bogota")-5*60*60

zoo12hr_epmaps <- zoo12hr_sum

epmaps_df12hr = data.frame(index(zoo12hr_epmaps),zoo12hr_epmaps)

epmaps_df12hr11072023 = epmaps_df12hr[,-2]

colnames(epmaps_df12hr11072023) =c("time",
"C02EPM","C04EPM","C05EPM","C14EPM","C15EPM","P03EPM","P08EPM","P09EPM",
"P11EPM","P12EPM","P27EPM","P28EPM","P31EPM","P32EPM","P70EPM")

save(epmaps_df12hr11072023, file="EPMAPSdf12hr15072023.RData")

save(zoo12hr_epmaps, file="EPMAPSz12hr15072023.RData")

#load("EPMAPSdf12hr11072023.RData")

#load("EPMAPSz12hr11072023.RData")

#-----

#hacer indice cada 24 horas

ind_24hr <- strptime("1970-01-01", "%Y-%m-%d") +
ceiling(as.numeric(index(epmaps_z11072023))/(24*60*60))*(24*60*60)-5*60*60 #sum 1-
5->5, 6-10 -> 10, 11-15->15, 16-20->20

zoo24hr_sum <- aggregate(epmaps_z11072023,ind_24hr,function(x) round(sum(x,na.rm
= F),1))

converted_index2 <- as.POSIXlt(index(zoo24hr_sum), tz = "America/Bogota")-5*60*60

zoo24hr_epmaps <- zoo24hr_sum

epmaps_df24hr = data.frame(index(zoo24hr_epmaps),zoo24hr_epmaps)

epmaps_df24hr11072023 = epmaps_df24hr[,-2]

colnames(epmaps_df24hr11072023) =c("time",
"C02EPM","C04EPM","C05EPM","C14EPM","C15EPM","P03EPM","P08EPM","P09EPM",
"P11EPM","P12EPM","P27EPM","P28EPM","P31EPM","P32EPM","P70EPM")

save(epmaps_df24hr11072023, file="EPMAPSdf24hr15072023.RData")

save(zoo24hr_epmaps, file="EPMAPSz24hr15072023.RData")

```

```

#load("EPMAPSdf24hr15072023.RData)
#load("EPMAPSz24hr15072023.RData")

#-----

#####

##4. Agrupar archivos pos estación INAMHI

#####

rm(list=ls())

#M0003 abrir y juntar las series temporales

m0003<-list.files(pattern = "M0003")

date_max_M0003<-as.character()

date_min_M0003<-as.character()

for (i in 1:length(m0003)){

  data = read.csv(m0003[i],header=T)

  a<-data

  a[,1]<-as.POSIXct(a[,1],format='%Y-%m-%d %H:%M:%S')

  a[,1]<-round_date(a[,1], unit="1 min")

  a[,2]= as.numeric(a[,2])

  #borrar duplicado

  borrar <- which(duplicated(a[,1]))

  if (length(borrar) > 0) {

    a <- a[-borrar,]

  }

  #crear zoo

  date_min_M0003[i]<-as.character(a[1,1])

  date_max_M0003[i]<-as.character(a[nrow(a),1])

```

```

a_z<-zoo(a[,2],a[,1])

assign(paste('z',m0003[i],sep='_'),a_z)

#crear data.frame

a_df=data.frame(index(a_z),coredata(a_z))

assign(paste('df',m0003[i],sep='_'),a_df)

}

m0003f = rbind.data.frame(df_M0003_2014.csv,df_M0003_2015.csv,

                        df_M0003_2016.csv,df_M0003_2017.csv,

                        df_M0003_2018.csv,df_M0003_2019.csv,

                        df_M0003_2020.csv,df_M0003_2021.csv,

                        df_M0003_2022.csv)

colnames(m0003f) = c("fecha","dato")

#save(m0003f, file="m003f_15072023.RData")

#load("m003f_15072023.RData")

#####

#M0006 abrir y juntar las series temporales

m0006<-list.files(pattern = "M0006")

date_max_M0006<-as.character()

date_min_M0006<-as.character()

for (i in 1:length(m0006)){

  data = read.csv(m0006[i],header=T)

  a<-data

  a[,1]<-as.POSIXct(a[,1],format='%Y-%m-%d %H:%M:%S')

  a[,1]<-round_date(a[,1], unit="1 min")

  a[,2]= as.numeric(a[,2])

```

```

#borrar duplicado

borrar <- which(duplicated(a[,1]))

if (length(borrar) > 0) {

  a <- a[-borrar,]

}

#crear zoo

date_min_M0006[i]<-as.character(a[1,1])

date_max_M0006[i]<-as.character(a[nrow(a),1])

a_z<-zoo(a[,2],a[,1])

assign(paste('z',m0006[i],sep='_'),a_z)

#crear data.frame

a_df=data.frame(index(a_z),coredata(a_z))

assign(paste('df',m0006[i],sep='_'),a_df)

}

m0006f = rbind.data.frame(df_M0006_2013.csv,df_M0006_2014.csv,

                        df_M0006_2015.csv,df_M0006_2016.csv,

                        df_M0006_2017.csv,df_M0006_2018.csv)

colnames(m0006f) = c("fecha","dato")

#save(m0006f, file="m006f_15072023.RData")

#load("m006f_15072023.RData")

#####

#M0007 abir y juntar las series temporales

m0007<-list.files(pattern = "M0007")

date_max_M0007<-as.character()

date_min_M0007<-as.character()

```



```

for (i in 1:length(m0007)){

  data = read.csv(m0007[i],header=T)

  a<-data

  a[,1]<-as.POSIXct(a[,1],format='%Y-%m-%d %H:%M:%S')

  a[,1]<-round_date(a[,1], unit="1 min")

  a[,2]= as.numeric(a[,2])

  #borrar duplicado

  borrar <- which(duplicated(a[,1]))

  if (length(borrar) > 0) {

    a <- a[-borrar,]

  }

  #crear zoo

  a_z<-zoo(a[,2],a[,1])

  date_min_M0007[i]<-as.character(a[1,1])

  date_max_M0007[i]<-as.character(a[nrow(a),1])

  assign(paste('z',m0007[i],sep='_'),a_z)

  #crear data.frame

  a_df=data.frame(index(a_z),coredata(a_z))

  assign(paste('df',m0007[i],sep='_'),a_df)

}

m0007f = rbind.data.frame(df_M0007_2015.csv,df_M0007_2016.csv)

colnames(m0007f) = c("fecha","dato")

#save(m0007f, file="m0007f_15072023.RData")

#load("m007f_15072023.RData")

#####

```

```

#M0025 abrir y juntar las series temporales

m0025<-list.files(pattern = "M0025")

date_max_M0025<-as.character()

date_min_M0025<-as.character()

for (i in 1:length(m0025)){

  data = read.csv(m0025[i],header=T)

  a<-data

  a[,1]<-as.POSIXct(a[,1],format='%Y-%m-%d %H:%M:%S')

  a[,1]<-round_date(a[,1], unit="1 min")

  a[,2]= as.numeric(a[,2])

  #borrar duplicado

  borrar <- which(duplicated(a[,1]))

  if (length(borrar) > 0) {

    a <- a[-borrar,]

  }

  #crear zoo

  date_min_M0025[i]<-as.character(a[1,1])

  date_max_M0025[i]<-as.character(a[nrow(a),1])

  a_z<-zoo(a[,2],a[,1])

  assign(paste('z',m0025[i],sep='_'),a_z)

  #crear data.frame

  a_df=data.frame(index(a_z),coredata(a_z))

  assign(paste('df',m0025[i],sep='_'),a_df)

}

m0025f = rbind.data.frame(df_M0025_2014.csv,df_M0025_2015.csv,

```

```

df_M0025_2017.csv,df_M0025_2018.csv)

colnames(m0025f) = c("fecha","dato")

#save(m0025f, file="m0025f_15072023.RData")

#load("m0025f_11072023.RData")

#####

#M0033 abrir y juntar las series temporales

m0033<-list.files(pattern = "M0033")

date_max_M0033<-as.character()

date_min_M0033<-as.character()

for (i in 1:length(m0033)){

  data = read.csv(m0033[i],header=T)

  a<-data

  a[,1]<-as.POSIXct(a[,1],format='%Y-%m-%d %H:%M:%S')

  a[,1]<-round_date(a[,1], unit="1 min")

  a[,2]= as.numeric(a[,2])

  #borrar duplicado

  borrar <- which(duplicated(a[,1]))

  if (length(borrar) > 0) {

    a <- a[-borrar,]

  }

  #crear zoo

  date_min_M0033[i]<-as.character(a[1,1])

  date_max_M0033[i]<-as.character(a[nrow(a),1])

  a_z<-zoo(a[,2],a[,1])

  assign(paste('z',m0033[i],sep='_'),a_z)

```

```

#crear data.frame

a_df=data.frame(index(a_z),coredata(a_z))

assign(paste('df',m0033[i],sep='_'),a_df)

}

m0033f =
rbind.data.frame(df_M0033_2013.csv,df_M0033_2014.csv,df_M0033_2015.csv,
                 df_M0033_2016.csv,df_M0033_2017.csv,
                 df_M0033_2018.csv)

colnames(m0033f) = c("fecha","dato")

#save(m0033f, file="m0033f_15072023.RData")

#load("m0033f_15072023.RData")

#####

##5. Unir estaciones en un solo data.frame

#####

#encontrar fecha inicio y fin

date_min =
c(date_min_M0003,date_min_M0006,date_min_M0007,date_min_M0025,date_min_M00
33)

date_max =
c(date_max_M0003,date_max_M0006,date_max_M0007,date_max_M0025,date_max_M
0033)

#load("date_min15072023.RData")

#load("date_max15072023.RData")

#load("m0033f_15072023.RData")

#load("m0025f_15072023.RData")

#load("m0007f_15072023.RData")

#load("m0006f_15072023.RData")

```

```

#load("m003f_15072023.RData")

#encontrar la fecha de inicio y final (menor y mayor)

start = as.POSIXct(min(date_min))

end = as.POSIXct(max(date_max))

#objeto zoo vacío

index<-seq(start,end,by="1 min")

dummy_z <- zoo(rep(NA,length(index)),index)

#transformar a zoo

z_m0003 = zoo(m0003f$dato,m0003f$fecha)

z_m0006 = zoo(m0006f$dato,m0006f$fecha)

z_m0007 = zoo(m0007f$dato,m0007f$fecha)

z_m0025 = zoo(m0025f$dato,m0025f$fecha)

z_m0033 = zoo(m0033f$dato,m0033f$fecha)

#merge objetos zoo

inamhi_z <- merge(dummy_z,z_m0003,z_m0006,z_m0007,z_m0025,z_m0033)

inamhi_df = as.data.frame(inamhi_z)

colnames(inamhi_df) =c("time","M003INA","M006INA","M007INA","M025INA","M033INA")

save(inamhi_z, file="inamhi_z25072023.RData")

save(inamhi_df, file="inamhi_df25072023.RData")

#####

##6. Agregación INAMHI

#####

#hacer indice cada 30 minutos

ind_30min <- strptime("1970-01-01", "%Y-%m-%d") +
ceiling(as.numeric(index(inamhi_z))/(30*60))*(30*60)-5*60*60 #sum 1-5->5, 6-10 -> 10,
11-15->15, 16-20->20

```

```

zoo30min_sum <- aggregate(inamhi_z,ind_30min,function(x) round(sum(x,na.rm = F),1))
zoo30min_inamhi <- zoo30min_sum
inamhi_df30min = data.frame(index(zoo30min_inamhi),zoo30min_inamhi )
inamhi_df30min = inamhi_df30min[,-2]
colnames(inamhi_df30min)
=c("time","M003INA","M006INA","M007INA","M025INA","M033INA")
#save(inamhi_df30min, file="inamhi_df30min20230710.RData")
#load("inamhi_df30min20230707.RData")
#-----
#####
##7. Verificación calidad 30 min
#####
#https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2009JD012097
#Shen et al., 2010
#(1) the extreme check
#The extreme check is conducted by comparison with the maximum of the daily
precipitation time series for the target month. Reports of daily precipitation at many
stations are collected at manned meteorological stations and go through manual quality
check. An hourly report is rejected if its value exceeds the monthly maximum of daily
precipitation.
#-----
x_m0003 = 84.5
#aplicamos
#M0003
inamhi_df30min$`M003INA`[inamhi_df30min$`M003INA`>x_m0003]=NA
summary(inamhi_df30min$`M003INA`)
#vacios

```

```

#gaps = inamhi_df30min[which(!is.na(inamhi_df30min$`IZOBAMBA-INA`)),1]

gapsM003_inicio = which(!is.na(inamhi_df30min$`M003INA`))[1]

gapsM003_fin = which(!is.na(inamhi_df30min$`M003INA`))[length(
which(!is.na(inamhi_df30min$`M003INA`)))]

totalM003 = length(inamhi_df30min$`M003INA`[gapsM003_inicio:gapsM003_fin])

gapsM003 =
length(which(is.na(inamhi_df30min$`M003INA`)[gapsM003_inicio:gapsM003_fin]))

PgapsM003 = gapsM003/totalM003*100

#-----

x_m0006 = 197.6

#aplicamos

#M0006

inamhi_df30min$`M006INA`[inamhi_df30min$`M006INA`>x_m0006]=NA

summary(inamhi_df30min$M006INA)

gapsM006_inicio = which(!is.na(inamhi_df30min$`M006INA`))[1]

gapsM006_fin = which(!is.na(inamhi_df30min$`M006INA`))[length(
which(!is.na(inamhi_df30min$`M006INA`)))]

totalM006 = length(inamhi_df30min$`M006INA`[gapsM006_inicio:gapsM006_fin])

gapsM006 =
length(which(is.na(inamhi_df30min$`M006INA`)[gapsM006_inicio:gapsM006_fin]))

PgapsM006 = gapsM006/totalM006*100

#-----

x_m0007 = 168.8

#aplicamos

#M0007

inamhi_df30min$`M007INA`[inamhi_df30min$`M007INA`>x_m0007]=NA

summary(inamhi_df30min$`M007INA`)

```

```

gapsM007_inicio = which(!is.na(inamhi_df30min$`M007INA`))[1]
gapsM007_fin = which(!is.na(inamhi_df30min$`M007INA`))[length(
which(!is.na(inamhi_df30min$`M007INA`)))]

totalM007 = length(inamhi_df30min$`M007INA`[gapsM007_inicio:gapsM007_fin])
gapsM007 =
length(which(is.na(inamhi_df30min$`M007INA`)[gapsM007_inicio:gapsM007_fin]))

PgapsM007 = gapsM007/totalM007*100

#-----

x_m0025 = 215.6/24

#aplicamos

#M0025

inamhi_df30min$M025INA[inamhi_df30min$M025INA>x_m0025]=NA

summary(inamhi_df30min$M025INA)

gapsM025_inicio = which(!is.na(inamhi_df30min$`M025INA`))[1]
gapsM025_fin = which(!is.na(inamhi_df30min$`M025INA`))[length(
which(!is.na(inamhi_df30min$`M025INA`)))]

totalM025 = length(inamhi_df30min$`M025INA`[gapsM025_inicio:gapsM025_fin])
gapsM025 =
length(which(is.na(inamhi_df30min$`M025INA`)[gapsM025_inicio:gapsM025_fin]))

PgapsM025 = gapsM025/totalM025*100

#-----

x_m0033 = 65.4

#aplicamos

#M0033

inamhi_df30min$`M033INA`[inamhi_df30min$`M033INA`>x_m0033]=NA

summary(inamhi_df30min$`M033INA`)

gapsM033_inicio = which(!is.na(inamhi_df30min$`M033INA`))[1]

```



```

gapsM033_fin = which(!is.na(inamhi_df30min$`M033INA`))[length(
which(!is.na(inamhi_df30min$`M033INA`)))]

totalM033 = length(inamhi_df30min$`M033INA`[gapsM033_inicio:gapsM033_fin])

gapsM033 =
length(which(is.na(inamhi_df30min$`M033INA`)[gapsM033_inicio:gapsM033_fin]))

PgapsM033 = gapsM033/totalM033*100

#####

#nuevo

inamhi_dfc30min = inamhi_df30min

#transformar a zoo

z_m0003c = zoo(inamhi_dfc30min$M003INA,inamhi_dfc30min$time)
z_m0006c = zoo(inamhi_dfc30min$M006INA,inamhi_dfc30min$time)
z_m0007c = zoo(inamhi_dfc30min$M007INA,inamhi_dfc30min$time)
z_m0025c = zoo(inamhi_dfc30min$M025INA,inamhi_dfc30min$time)
z_m0033c = zoo(inamhi_dfc30min$M033INA,inamhi_dfc30min$time)

#merge objetos zoo

inamhi_zc <- merge (z_m0003c,z_m0006c,z_m0007c,z_m0025c,z_m0033c)

inamhi_dfc = as.data.frame(inamhi_zc)

colnames(inamhi_dfc) =c("M003INA","M006INA","M007INA","M025INA","M033INA")

save(inamhi_zc, file="inamhi_z30min20230726.RData")

save(inamhi_dfc, file="inamhi_df30min20230726.RData")

#load("inamhi_z30min20230715.RData")

#load("inamhi_df30min20230715.RData")

#-----

#hacer indice cada hora

```

```

ind_1hr <- strptime("1970-01-01", "%Y-%m-%d") +
ceiling(as.numeric(index(inamhi_zc))/(1*60*60))*(1*60*60)-5*60*60 #sum 1-5->5, 6-10 ->
10, 11-15->15, 16-20->20

zoo1hr_sum <- aggregate(inamhi_zc,ind_1hr,function(x) round(sum(x,na.rm = F),1))

zoo1hr_inamhi <- zoo1hr_sum

inamhi_df1hr = data.frame(index(zoo1hr_inamhi),zoo1hr_inamhi )

colnames(inamhi_df1hr)
=c("time","M003INA","M006INA","M007INA","M025INA","M033INA")

save(zoo1hr_inamhi, file="inamhi_z1hr20230726.RData")

save(inamhi_df1hr, file="inamhi_df1hr20230726.RData")

#load("inamhi_z1hr20230715.RData")

#load("inamhi_df1hr20230715.RData")

#-----

#hacer indice cada 2 horas

ind_2hr <- strptime("1970-01-01", "%Y-%m-%d") +
ceiling(as.numeric(index(inamhi_zc))/(2*60*60))*(2*60*60)-5*60*60 #sum 1-5->5, 6-10 ->
10, 11-15->15, 16-20->20

zoo2hr_sum <- aggregate(inamhi_zc,ind_2hr,function(x) round(sum(x,na.rm = F),1))

zoo2hr_inamhi <- zoo2hr_sum

inamhi_df2hr = data.frame(index(zoo2hr_inamhi),zoo2hr_inamhi )

colnames(inamhi_df2hr)
=c("time","M003INA","M006INA","M007INA","M025INA","M033INA")

save(zoo2hr_inamhi, file="inamhi_z2hr20230726.RData")

save(inamhi_df2hr, file="inamhi_df2hr20230726.RData")

#load("inamhi_df2hr20230707.RData")

#-----

#hacer indice cada 6 horas

```

```

ind_6hr <- strptime("1970-01-01", "%Y-%m-%d") +
ceiling(as.numeric(index(inamhi_zc))/(6*60*60))*(6*60*60)-5*60*60 #sum 1-5->5, 6-10 ->
10, 11-15->15, 16-20->20

zoo6hr_sum <- aggregate(inamhi_zc,ind_6hr,function(x) round(sum(x,na.rm = F),1))

zoo6hr_inamhi <- zoo6hr_sum

inamhi_df6hr = data.frame(index(zoo6hr_inamhi),zoo6hr_inamhi )

colnames(inamhi_df6hr)
=c("time","M003INA","M006INA","M007INA","M025INA","M033INA")

save(zoo6hr_inamhi, file="inamhi_z6hr20230726.RData")

save(inamhi_df6hr, file="inamhi_df6hr20230726.RData")

#laod("inamhi_df6hr20230707.RData")

#-----

#hacer indice cada 12 horas

ind_12hr <- strptime("1970-01-01", "%Y-%m-%d") +
ceiling(as.numeric(index(inamhi_zc))/(12*60*60))*(12*60*60)-5*60*60 #sum 1-5->5, 6-10 -
> 10, 11-15->15, 16-20->20

zoo12hr_sum <- aggregate(inamhi_zc,ind_12hr,function(x) round(sum(x,na.rm = F),1))

zoo12hr_inamhi <- zoo12hr_sum

inamhi_df12hr = data.frame(index(zoo12hr_inamhi),zoo12hr_inamhi )

colnames(inamhi_df12hr)
=c("time","M003INA","M006INA","M007INA","M025INA","M033INA")

save(zoo12hr_inamhi, file="inamhi_z12hr20230726.RData")

save(inamhi_df12hr, file="inamhi_df12hr20230726.RData")

#laod("inamhi_df12hr20230707.RData")

#-----

#hacer indice cada 24 horas

```

```

ind_24hr <- strptime("1970-01-01", "%Y-%m-%d") +
ceiling(as.numeric(index(inamhi_zc))/(24*60*60))*(24*60*60)-5*60*60 #sum 1-5->5, 6-10 -
> 10, 11-15->15, 16-20->20

zoo24hr_sum <- aggregate(inamhi_zc,ind_24hr,function(x) round(sum(x,na.rm = F),1))

zoo24hr_inamhi <- zoo24hr_sum

inamhi_df24hr = data.frame(index(zoo24hr_inamhi),zoo24hr_inamhi )

colnames(inamhi_df24hr)
=c("time","M003INA","M006INA","M007INA","M025INA","M033INA")

save(zoo24hr_inamhi, file="inamhi_z24hr20230726.RData")

save(inamhi_df24hr, file="inamhi_df24hr20230726.RData")

#load("inamhi_df24hr20230707.RData")

#-----

```

Anexo 4. Script de gráficas de series temporales

```

#----- TIC ISPF -----

#Grafico ST

library(zoo)

library(reshape)

library(ggplot2)

library(ggpubr)

library(dplyr)

rm(list=ls())

#####

# 1.- CARGAR DATOS

#####

#EPMAPS cargar datos

#load("EPMAPSz1hr15072023.RData")

```

```

#load("EPMAPSz2hr15072023.RData")
#load("EPMAPSz6hr15072023.RData")
#load("EPMAPSz12hr15072023.RData")
#load("EPMAPSz24hr15072023.RData")
load("EPMAPSz30min15072023.RData")
#INAMHI cargar datos
#load("inamhi_z1hr20230726.RData")
#load("inamhi_z2hr20230726.RData")
#load("inamhi_z12hr20230726.RData")
#load("inamhi_z24hr20230726.RData")
load("inamhi_z30min20230726.RData")
#load("inamhi_z6hr20230726.RData")
#IMERG cargar datos
load("open_z14072023.RData")
#load("open_z1hr14072023.RData")
#load("open_z2hr14072023.RData")
#load("open_z6hr14072023.RData")
#load("open_z12hr14072023.RData")
#load("open_z24hr14072023.RData")

#####

# 2.- UNIR DATOS

#####

#30MINUTOS

totaldata_z30min = merge(zoo30min_epmaps,inamhi_zc,open_z11072023)

totaldata_df30min = as.data.frame(totaldata_z30min)

```

```

#save(totaldata_z30min,file = "Totalz30min26072023.RData")

#1hora

#totaldata_z1hr = merge(zoo1hr_epmaps,zoo1hr_inamhi,zoo1hr_open)

#totaldata_df1hr = as.data.frame(totaldata_z1hr)

#save(totaldata_z1hr,file = "Totalz1hr26072023.RData")

#2hora

#totaldata_z2hr = merge(zoo2hr_epmaps,zoo2hr_inamhi,zoo2hr_open)

#totaldata_df2hr = as.data.frame(totaldata_z2hr)

#save(totaldata_z2hr,file = "Totalz2hr26072023.RData")

#6hora

#totaldata_z6hr= merge(zoo6hr_epmaps,zoo6hr_inamhi,zoo6hr_open)

#totaldata_df6hr = as.data.frame(totaldata_z6hr)

#save(totaldata_z6hr,file = "Totalz6hr26072023.RData")

#12hora

#totaldata_z12hr = merge(zoo12hr_epmaps,zoo12hr_inamhi,zoo12hr_open)

#totaldata_df12hr = as.data.frame(totaldata_z12hr)

#save(totaldata_z12hr,file = "Totalz12hr26072023.RData")

#24hora

#totaldata_z24hr = merge(zoo24hr_epmaps,zoo24hr_inamhi,zoo24hr_open)

#totaldata_df24hr = as.data.frame(totaldata_z24hr)

#save(totaldata_z24hr,file = "Totalz24hr26072023.RData")

data30min = totaldata_df30min[,-c(1,22)]

#data1hr = totaldata_df1hr[,-c(1,22)]

#data2hr = totaldata_df2hr[,-c(1,22)]

```

```

#data6hr = totaldata_df6hr[,-c(1,22)]

#data12hr = totaldata_df12hr[,-c(1,22)]

#data24hr = totaldata_df24hr[,-c(1,22)]

#####

# 3.- GRFICOS

#####

##C15 vs IMERG

#seleccionamos las columnas y agregamos a mes

data_nan = cbind.data.frame(as.POSIXct(rownames(data30min), format="%Y-%m-%d")
,data30min$z_EPMAPS_C15_NANEGALITO,data30min$C15_z)

colnames(data_nan) = c("date","C15epmaps","C15imerg")

data_nan$date15 = format(as.Date(data_nan$date), "%Y-%m")

c15epmaps = aggregate(C15epmaps ~ date15, data_nan, sum)

c15imerg = aggregate(C15imerg ~ date15, data_nan, sum)

index= seq(as.POSIXct(paste(c15imerg$date15[1],"01",sep="-"), format="%Y-%m-%d"),
           as.POSIXct(paste(c15imerg$date15[nrow(c15imerg)],"01",sep="-"), format="%Y-
%m-%d"), by="month")

index2= substr(as.character(index),1,7)

ind0 =NA

for (i in 1:nrow(c15epmaps)) {

  ind = which(c15epmaps$date15[i] == index2)

  ind0 = c(ind0,ind)

}

ind0 = ind0[-1]

c15_mes = c15imerg

c15_mes$epmaps = NA

```

```

c15_mes$epmaps[ind0] = c15epmaps$C15epmaps

c15_mes$date15 = as.POSIXct(paste(c15_mes$date15,"01",sep="-"), format="%Y-%m-
%d")

colnames(c15_mes)=c("date15","IMERG V06","OBSERVADO")

#plot

melt_c15mes = melt(c15_mes, id="date15")

plt_c15mes = ggplot(melt_c15mes, aes(x=date15, y=value, color=variable))+

  geom_line()+

  theme_bw() +

  labs (title="C15", x= NULL, y="Precipitación [mm])+

  theme(plot.title = element_text(size = 20), legend.position = "none",

        legend.background = element_rect(fill = "#FFBEB2", colour = "black"),

        legend.text = element_text(size=14), legend.title = element_blank(),

        axis.title.x = element_text(size=16), axis.title.y = element_text(size=12),

        axis.text.y = element_text(size=12), axis.text.x = element_text(size=14))+

  scale_color_manual(values = c("blue","red"))

#-----

##M003 vs IMERG

#seleccionamos las columnas y agregamos a mes

data_nan3 = cbind.data.frame(as.POSIXct(rownames(data30min), format="%Y-%m-%d")
,data30min$z_m0003c,data30min$M003_z)

colnames(data_nan3) = c("date","M003inamhi","M003imerg")

data_nan3$date3 = format(as.Date(data_nan3$date), "%Y-%m")

M003inamhi = aggregate(M003inamhi ~ date3, data_nan3, sum)[-c(75:85),]

M003imerg = aggregate(M003imerg ~ date3, data_nan3, sum)

index3= seq(as.POSIXct(paste(M003imerg$date3[1],"01",sep="-"), format="%Y-%m-%d"),

```



```

as.POSIXct(paste(M003imerg$date3[nrow(M003imerg)],"01",sep="-"),
format="%Y-%m-%d"), by="month")

index32= substr(as.character(index3),1,7)

ind30 =NA

for (i in 1:nrow(M003inamhi)) {

  ind3 = which(M003inamhi$date3[i] == index32)

  ind30 = c(ind30,ind3)

}

ind30 = ind30[-1]

M003_mes = M003imerg

M003_mes$inamhi = NA

M003_mes$inamhi[ind30] = M003inamhi$M003inamhi

M003_mes$date3 = as.POSIXct(paste(M003_mes$date3,"01",sep="-"), format="%Y-%m-
%d")

colnames(M003_mes)=c("date3","IMERG V06","OBSERVADO")

#plot

melt_M003mes = melt(M003_mes, id="date3")

plt_M003mes = ggplot(melt_M003mes, aes(x=date3, y=value, color=variable))+

  geom_line()+

  theme_bw() +

  labs (title="M003", x= NULL, y="Precipitación [mm]")+

  theme(plot.title = element_text(size = 20), legend.position = "none",

        legend.background = element_rect(fill = "#FFBEB2", colour = "black"),

        legend.text = element_text(size=14), legend.title = element_blank(),

        axis.title.x = element_text(size=16), axis.title.y = element_text(size=12),

        axis.text.y = element_text(size=12), axis.text.x = element_text(size=14))+

```

```

scale_color_manual(values = c("blue","red"))

#-----

###M006 vs IMERG

#seleccionamos las columnas y agregamos a mes

data_nan6 = cbind.data.frame(as.POSIXct(rownames(data30min), format="%Y-%m-%d"),
,data30min$z_m0006c,data30min$M006_z)

colnames(data_nan6) = c("date","M006inamhi","M006imerg")

data_nan6$date6 = format(as.Date(data_nan6$date), "%Y-%m")

M006inamhi = aggregate(M006inamhi ~ date6, data_nan6, sum)

M006imerg = aggregate(M006imerg ~ date6, data_nan6, sum)

index6= seq(as.POSIXct(paste(M006imerg$date6[1],"01",sep="-"), format="%Y-%m-%d"),
            as.POSIXct(paste(M006imerg$date6[nrow(M006imerg)],"01",sep="-"),
            format="%Y-%m-%d"), by="month")

index62= substr(as.character(index6),1,7)

ind60 =NA

for (i in 1:nrow(M006inamhi)) {

  ind = which(M006inamhi$date6[i] == index62)

  ind60 = c(ind60,ind)

}

ind60 = ind60[-1]

M006_mes = M006imerg

M006_mes$inamhi = NA

M006_mes$inamhi[ind60] = M006inamhi$M006inamhi

M006_mes$date6 = as.POSIXct(paste(M006_mes$date6,"01",sep="-"), format="%Y-%m-%d")

```

```

colnames(M006_mes)=c("date6","IMERG V06","OBSERVADO")

#plot

melt_M006mes = melt(M006_mes, id="date6")

plt_M006mes = ggplot(melt_M006mes, aes(x=date6, y=value, color=variable))+

geom_line()+

theme_bw() +

labs (title="M006", x= NULL, y="Precipitación [mm]")+

theme(plot.title = element_text(size = 20), legend.position = "none",

      legend.background = element_rect(fill = "#FFBEB2", colour = "black"),

      legend.text = element_text(size=14), legend.title = element_blank(),

      axis.title.x = element_text(size=16), axis.title.y = element_text(size=12),

      axis.text.y = element_text(size=12), axis.text.x = element_text(size=14))+

scale_color_manual(values = c("blue","red"))

#-----

# M007 vs IMERG

#seleccionamos las columnas y agregamos a mes

data_nan7 = cbind.data.frame(as.POSIXct(rownames(data30min), format="%Y-%m-%d")
,data30min$z_m0007c,data30min$M007_z)

colnames(data_nan7) = c("date","M007inamhi","M007imerg")

data_nan7$date7 = format(as.Date(data_nan7$date), "%Y-%m")

M007inamhi = aggregate(M007inamhi ~ date7, data_nan7, sum)

M007imerg = aggregate(M007imerg ~ date7, data_nan7, sum)

index7= seq(as.POSIXct(paste(M007imerg$date7[1],"01",sep="-"), format="%Y-%m-%d"),

           as.POSIXct(paste(M007imerg$date7[nrow(M007imerg)],"01",sep="-"),

           format="%Y-%m-%d"), by="month")

index72= substr(as.character(index7),1,7)

```

```

ind70 =NA

for (i in 1:nrow(M007inamhi)) {

  ind = which(M007inamhi$date7[i] == index72)

  ind70 = c(ind70,ind)

}

ind70 = ind70[-1]

M007_mes = M007imerg

M007_mes$inamhi = NA

M007_mes$inamhi[ind70] = M007inamhi$M007inamhi

M007_mes$date7 = as.POSIXct(paste(M007_mes$date7,"01",sep="-"), format="%Y-%m-
%d")

colnames(M007_mes)=c("date7","IMERG V06","OBSERVADO")

#plot

melt_M007mes = melt(M007_mes, id="date7")

plt_M007mes = ggplot(melt_M007mes, aes(x=date7, y=value, color=variable))+

  geom_line()+

  theme_bw() +

  labs (title="M007", x= NULL, y="Precipitación [mm]")+

  theme(plot.title = element_text(size = 20), legend.position = "none",

        legend.background = element_rect(fill = "#FFBEB2", colour = "black"),# Borde

        legend.text = element_text(size=14), legend.title = element_blank(),

        axis.title.x = element_text(size=16), axis.title.y = element_text(size=12),

        axis.text.y = element_text(size=12), axis.text.x = element_text(size=14))+

  scale_color_manual(values = c("blue","red"))

#-----

# M025 vs IMERG

```

```

#seleccionamos las columnas y agregamos a mes

data_nan25 = cbind.data.frame(as.POSIXct(rownames(data30min), format="%Y-%m-
%d") ,data30min$z_m0025c,data30min$M025_z)

colnames(data_nan25) = c("date","M025inamhi","M025imerg")

data_nan25$date25 = format(as.Date(data_nan25$date), "%Y-%m")

M025inamhi = aggregate(M025inamhi ~ date25, data_nan25, sum)

M025imerg = aggregate(M025imerg ~ date25, data_nan25, sum)

index25= seq(as.POSIXct(paste(M025imerg$date25[1],"01",sep="-"), format="%Y-%m-
%d"),

              as.POSIXct(paste(M025imerg$date25[nrow(M025imerg)],"01",sep="-"),
format="%Y-%m-%d"), by="month")

index225= substr(as.character(index25),1,7)

ind250 =NA

for (i in 1:nrow(M025inamhi)) {

  ind = which(M025inamhi$date25[i] == index225)

  ind250 = c(ind250,ind)

}

ind250 = ind250[-1]

M025_mes = M025imerg

M025_mes$inamhi = NA

M025_mes$inamhi[ind250] = M025inamhi$M025inamhi

M025_mes$date25 = as.POSIXct(paste(M025_mes$date25,"01",sep="-"), format="%Y-
%m-%d")

colnames(M025_mes)=c("date25","IMERG V06","OBSERVADO")

#plot

melt_M025mes = melt(M025_mes, id="date25")

plt_M025mes = ggplot(melt_M025mes, aes(x=date25, y=value, color=variable))+

```

```

geom_line()+
theme_bw() +
labs (title="M025", x= NULL, y="Precipitación [mm]")+
theme(plot.title = element_text(size = 20), legend.position = "none",
      legend.text = element_text(size=14), legend.title = element_blank(),
      legend.background = element_rect(fill = "#FFBEB2", colour = "black"),
      axis.title.x = element_text(size=16), axis.title.y = element_text(size=12),
      axis.text.y = element_text(size=12), axis.text.x = element_text(size=14))+
scale_color_manual(values = c("blue","red"))

#-----
#M033 vs IMERG

#seleccionamos las columnas y agregamos a mes

data_nan33 = cbind.data.frame(as.POSIXct(rownames(data30min), format="%Y-%m-
%d") ,data30min$z_m0033c,data30min$M033_z)

colnames(data_nan33) = c("date","M033inamhi","M033imerg")

data_nan33$date33 = format(as.Date(data_nan33$date), "%Y-%m")

M033inamhi = aggregate(M033inamhi ~ date33, data_nan33, sum)

M033imerg = aggregate(M033imerg ~ date33, data_nan33, sum)

index33= seq(as.POSIXct(paste(M033imerg$date33[1],"01",sep="-"), format="%Y-%m-
%d"),

              as.POSIXct(paste(M033imerg$date33[nrow(M033imerg)],"01",sep="-"),
format="%Y-%m-%d"), by="month")

index332= substr(as.character(index33),1,7)

ind330 =NA

for (i in 1:nrow(M033inamhi)) {

  ind = which(M033inamhi$date33[i] == index332)

```

```

ind330 = c(ind330,ind)
}
ind330 = ind330[-1]
M033_mes = M033imerg
M033_mes$inamhi = NA
M033_mes$inamhi[ind330] = M033inamhi$M033inamhi
M033_mes$date33 = as.POSIXct(paste(M033_mes$date33,"01",sep="-"), format="%Y-
%m-%d")
colnames(M033_mes)=c("date33","IMERG V06","OBSERVADO")
#plot
melt_M033mes = melt(M033_mes, id="date33")
plt_M033mes = ggplot(melt_M033mes, aes(x=date33, y=value, color=variable))+
  geom_line()+
  theme_bw() +
  labs (title="M033", x= NULL, y="Precipitación [mm]")+
  theme(plot.title = element_text(size = 20), legend.position = "none",
        legend.background = element_rect(fill = "#FFBEB2", colour = "black"),
        legend.text = element_text(size=14), legend.title = element_blank(),
        axis.title.x = element_text(size=16), axis.title.y = element_text(size=12),
        axis.text.y = element_text(size=12), axis.text.x = element_text(size=14))+
  scale_color_manual(values = c("blue","red"))
#-----
##P70+P09+C05 vs IMERG
#seleccionamos las columnas y agregamos a mes
data_3 = cbind.data.frame(as.POSIXct(rownames(data30min), format="%Y-%m-%d")
, data30min$z_EPMAPS_C05_BELLAVISTA, data30min$z_EPMAPS_P09_INAQUITO_IN

```

```

AMHI,data30min$z_EPMAPS_P70_CC_EL_BOSQUE,data30min$P70_z,data30min$P09
_z)

colnames(data_3) = c("date","c05","p09","p70","p09i","p70i")

data_3$date10 = format(as.Date(data_3$date), "%Y-%m")

c05_mes = aggregate(c05 ~ date10, data_3, sum)[-c(1,215:219),]

p09_mes = aggregate(p09 ~ date10, data_3, sum)[-c(1,218),]

p70_mes = aggregate(p70 ~ date10, data_3, sum)[-c(48,49),]

p09i_mes = aggregate(p09i ~ date10, data_3, sum)

p70i_mes = aggregate(p70i ~ date10, data_3, sum)

index= seq(as.POSIXct(paste(p09i_mes$date10[1],"01",sep="-"), format="%Y-%m-%d"),
           as.POSIXct(paste(p09i_mes$date10[nrow(p09i_mes)],"01",sep="-"), format="%Y-
%m-%d"), by="month")

index2= substr(as.character(index),1,7)

#encontrar indices c05

ind_c050 =NA

for (i in 1:nrow(c05_mes)) {

  ind = which(c05_mes$date10[i] == index2)

  ind_c050 = c(ind_c050,ind)

}

ind_c050 = ind_c050[-1]

#encontrar indices p09

ind_p090 =NA

for (i in 1:nrow(p09_mes)) {

  ind = which(p09_mes$date10[i] == index2)

  ind_p090 = c(ind_p090,ind)

}

```



```

ind_p090 = ind_p090[-1]

#encontrar indices p70

ind_p700 =NA

for (i in 1:nrow(p70_mes)) {

  ind = which(p70_mes$date10[i] == index2)

  ind_p700 = c(ind_p700,ind)

}

ind_p700 = ind_p700[-1]

px09i_mes = p09i_mes

px09i_mes$c05 = NA

px09i_mes$p09 = NA

px09i_mes$p70 = NA

px09i_mes$p70i = NA

px09i_mes$c05[ind_c050] = c05_mes$c05

px09i_mes$p09[ind_p090] = p09_mes$p09

px09i_mes$p70[ind_p700] = p70_mes$p70

px09i_mes$p70i = p70i_mes$p70i

px09i_mes = px09i_mes[,-6]

px09i_mes$date10 = as.POSIXct(paste(px09i_mes$date10,"01",sep="-"), format="%Y-
%m-%d")

#plot

melt_px09mes = melt(px09i_mes, id="date10")

plt_px09mes = ggplot(melt_px09mes, aes(x=date10, y=value, color=variable))+

  geom_line()+

  theme_bw() +

```

```

labs (title="C05, P09, P70", x= NULL, y="Precipitación [mm]")+
theme(plot.title = element_text(size = 20), legend.position = "none",
      legend.text = element_text(size=14), legend.title = element_blank(),
      legend.background = element_rect(fill = "#FFBEB2", colour = "black"),
      axis.title.x = element_text(size=16), axis.title.y = element_text(size=12),
      axis.text.y = element_text(size=12), axis.text.x = element_text(size=14))+
scale_color_manual(values = c("blue", "red", "#C71F3C", "#AE123A"))

#-----
###P27+P12 vs IMERG

#seleccionamos las columnas y agregamos a mes

data_5 = cbind.data.frame(as.POSIXct(rownames(data30min), format="%Y-%m-%d")
, data30min$z_EPMAPS_P12_TOCTIUOCO_Pr, data30min$z_EPMAPS_P27_SAN_FRAN
CISCO, data30min$P12_z, data30min$P27_z)

colnames(data_5) = c("date", "P12", "P27", "P27i", "Imerg")

data_5$date2 = format(as.Date(data_5$date), "%Y-%m")

P12_mes = aggregate(P12 ~ date2, data_5, sum)[-c(224:228),]

P27_mes = aggregate(P27 ~ date2, data_5, sum)[-c(189:191),]

P27i_mes = aggregate(P27i ~ date2, data_5, sum)

index= seq(as.POSIXct(paste(P27i_mes$date2[1], "01", sep="-"), format="%Y-%m-%d"),
          as.POSIXct(paste(P27i_mes$date2[nrow(P27i_mes)], "01", sep="-"), format="%Y-
%m-%d"), by="month")

index2= substr(as.character(index), 1, 7)

#encontrar indices P12

ind_P120 =NA

for (i in 1:nrow(P12_mes)) {

```

```

ind = which(P12_mes$date2[i] == index2)

ind_P120 = c(ind_P120,ind)

}

ind_P120 = ind_P120[-1]

#encontrar indices p27

ind_P270 =NA

for (i in 1:nrow(P27_mes)) {

  ind = which(P27_mes$date2[i] == index2)

  ind_P270 = c(ind_P270,ind)

}

ind_P270 = ind_P270[-1]

Px27i_mes = P27i_mes

Px27i_mes$P12 = NA

Px27i_mes$P27 = NA

Px27i_mes$P27i = NA

Px27i_mes$P12[ind_P120] = P12_mes$P12

Px27i_mes$P27[ind_P270] = P27_mes$P27

Px27i_mes$P27i = P27i_mes$P27i

Px27i_mes = Px27i_mes#[,-2]

Px27i_mes$date2 = as.POSIXct(paste(Px27i_mes$date2,"01",sep="-"), format="%Y-%m-
%d")

#plot

melt_Px27mes = melt(Px27i_mes, id="date2")

plt_Px27mes = ggplot(melt_Px27mes, aes(x=date2, y=value, color=variable))+

  geom_line()+

  theme_bw() +

```

```

labs (title="P12, P27", x= NULL, y="Precipitación [mm]")+
theme(plot.title = element_text(size = 20), legend.position = "none",
      legend.text = element_text(size=14), legend.title = element_blank(),
      legend.background = element_rect(fill = "#FFBEB2", colour = "black"),
      axis.title.x = element_text(size=16), axis.title.y = element_text(size=12),
      axis.text.y = element_text(size=12), axis.text.x = element_text(size=14))+
scale_color_manual(values = c("blue","red","#AE123A"))

#-----

##P03+P08+P11+P28+P31+P32+C02+C04+C14 vs IMERG

#seleccionamos las columnas y agregamos a mes

data_100 = cbind.data.frame(as.POSIXct(rownames(data30min), format="%Y-%m-%d")
, data30min$z_EPMAPS_C02_RUMIHURCO, data30min$z_EPMAPS_C04_RUMIPAMBA,
data30min$z_EPMAPS_C14_MINDO_CAPTACION, data30min$z_EPMAPS_P03_RUMI
HURCO_MACHANGARA, data30min$z_EPMAPS_P08_RUMIPAMBA_BODEGAS, data30
min$z_EPMAPS_P11_ANTENAS, data30min$z_EPMAPS_P32_MINDO_BAJO, data30mi
n$z_EPMAPS_P28_CRUZ_LOMA, data30min$z_EPMAPS_P31_PICHAN, data30min$C0
2_z)

colnames(data_100) =
c("date", "C02", "C04", "C14", "P03", "P08", "P11", "P32", "P28", "P31", "C02i")

data_100$date2 = format(as.Date(data_100$date), "%Y-%m")

C02_mes = aggregate(C02 ~ date2, data_100, sum)[-c(205),]
C04_mes = aggregate(C04 ~ date2, data_100, sum)[-c(199:204),]
C14_mes = aggregate(C14 ~ date2, data_100, sum)[-c(163),]
P03_mes = aggregate(P03 ~ date2, data_100, sum)[-c(219:221),]
P08_mes = aggregate(P08 ~ date2, data_100, sum)[-c(222:224),]
P11_mes = aggregate(P11 ~ date2, data_100, sum)[-c(220:223),]
P32_mes = aggregate(P32 ~ date2, data_100, sum)
P28_mes = aggregate(P28 ~ date2, data_100, sum)[-c(201:202),]

```

```

P31_mes = aggregate(P31 ~ date2, data_100, sum)[-c(183:186),]

C02i_mes = aggregate(C02i ~ date2, data_100, sum)

index100= seq(as.POSIXct(paste(C02i_mes$date2[1],"01",sep="-"), format="%Y-%m-
%d"),

              as.POSIXct(paste(C02i_mes$date2[nrow(C02i_mes)],"01",sep="-"), format="%Y-
%m-%d"), by="month")

index1002= substr(as.character(index100),1,7)

#encontrar indices C04

ind_C040 =NA

for (i in 1:nrow(C04_mes)) {

  ind = which(C04_mes$date2[i] == index1002)

  ind_C040 = c(ind_C040,ind)

}

ind_C040 = ind_C040[-1]

#encontrar indices C14

ind_C140 =NA

for (i in 1:nrow(C14_mes)) {

  ind = which(C14_mes$date2[i] == index1002)

  ind_C140 = c(ind_C140,ind)

}

ind_C140 = ind_C140[-1]

#encontrar indices C02

ind_C020 =NA

for (i in 1:nrow(C02_mes)) {

  ind = which(C02_mes$date2[i] == index1002)

  ind_C020 = c(ind_C020,ind)

```

```

}

ind_C020 = ind_C020[-1]

#encontrar indices P03

ind_P030 =NA

for (i in 1:nrow(P03_mes)) {

  ind = which(P03_mes$date2[i] == index1002)

  ind_P030 = c(ind_P030,ind)

}

ind_P030 = ind_P030[-1]

#encontrar indices P08

ind_P080 =NA

for (i in 1:nrow(P08_mes)) {

  ind = which(P08_mes$date2[i] == index1002)

  ind_P080 = c(ind_P080,ind)

}

ind_P080 = ind_P080[-1]

#encontrar indices P11

ind_P110 =NA

for (i in 1:nrow(P11_mes)) {

  ind = which(P11_mes$date2[i] == index1002)

  ind_P110 = c(ind_P110,ind)

}

ind_P110 = ind_P110[-1]

#encontrar indices P32

```

```

ind_P320 =NA
for (i in 1:nrow(P32_mes)) {
  ind = which(P32_mes$date2[i] == index1002)
  ind_P320 = c(ind_P320,ind)
}
ind_P320 = ind_P320[-1]
#encontrar indices P28
ind_P280 =NA
for (i in 1:nrow(P28_mes)) {
  ind = which(P28_mes$date2[i] == index1002)
  ind_P280 = c(ind_P280,ind)
}
ind_P280 = ind_P280[-1]
#encontrar indices P31
ind_P310 =NA
for (i in 1:nrow(P31_mes)) {
  ind = which(P31_mes$date2[i] == index1002)
  ind_P310 = c(ind_P310,ind)
}
ind_P310 = ind_P310[-1]
Cx02i_mes = C02i_mes
Cx02i_mes$C02 = NA
Cx02i_mes$C04 = NA
Cx02i_mes$C14 = NA
Cx02i_mes$P03 = NA

```

```

Cx02i_mes$P08 = NA
Cx02i_mes$P11 = NA
Cx02i_mes$P32 = NA
Cx02i_mes$P28 = NA
Cx02i_mes$P31 = NA
Cx02i_mes$C02i = NA
Cx02i_mes$C02[ind_C020] = C02_mes$C02
Cx02i_mes$C04[ind_C040] = C04_mes$C04
Cx02i_mes$C14[ind_C140] = C14_mes$C14
Cx02i_mes$P03[ind_P030] = P03_mes$P03
Cx02i_mes$P08[ind_P080] = P08_mes$P08
Cx02i_mes$P11[ind_P110] = P11_mes$P11
Cx02i_mes$P32[ind_P320] = P32_mes$P32
Cx02i_mes$P28[ind_P280] = P28_mes$P28
Cx02i_mes$P31[ind_P310] = P31_mes$P31
Cx02i_mes$C02i = C02i_mes$C02
Cx02i_mes = Cx02i_mes#[,-2]
Cx02i_mes$date2 = as.POSIXct(paste(Cx02i_mes$date2,"01",sep="-"), format="%Y-%m-
%d")
#plot
melt_Cx02mes = melt(Cx02i_mes, id="date2")
plt_Cx02mes = ggplot(melt_Cx02mes, aes(x=date2, y=value, color=variable))+
  geom_line()+
  theme_bw() +
  labs (title="C02, C04, C14, P03, P08, P11, P28, P31, P32", x= NULL, y="Precipitación
[mm]")+

```



```

theme(plot.title = element_text(size = 17), legend.position = "none",
      legend.background = element_rect(fill = "#FFBEB2", colour = "black"),
      legend.text = element_text(size=14), legend.title = element_blank(),
      axis.title.x = element_text(size=16), axis.title.y = element_text(size=12),
      axis.text.y = element_text(size=12), axis.text.x = element_text(size=14))+

scale_color_manual(values =
c("blue", "red", "#FFBEB2", "#FA937E", "#F5725E", "#EB4F48", "#D72E3E", "#CC243C", "#BB
173A", "#AE123A"))

#####

#unir plots

plt_fig2 <- ggarrange(plt_px09mes, plt_Px27mes, plt_M003mes, plt_M006mes,
                    common.legend = F, legend = "none",
                    ncol = 2, nrow = 2, align = "v")

png(file='fig111_plots.png', width = 900, height = 500)

plt_fig2

dev.off()

plt_fig3 <- ggarrange(plt_M007mes, plt_M025mes, plt_M033mes, plt_c15mes,
                    common.legend = T, legend = "bottom",
                    ncol = 2, nrow = 2, align = "v")

png(file='fig112_plots.png', width = 900, height = 500)

plt_fig3

dev.off()

plt_fig4 <- ggarrange(plt_Cx02mes,
                    common.legend = F, legend = "none",
                    ncol = 1, nrow = 1, align = "v")

png(file='fig113_plots.png', width = 900, height = 250)

```

plt_fig4

dev.off()

#####

Anexo 5. Script de la validación de los datos

#----- TIC SPP IDFs -----

#----- Análisis AMS -----

library(lmomco)

library(lubridate)

library(openxlsx)

library(minpack.lm)

library(investr)

library(dplyr)

library(xts)

library(sp)

library(rgdal)

library(Metrics)

library(hydroGOF)

library(zoo)

library(reshape)

library(ggplot2)

library(ggpubr)

rm(list=ls())

#####

1.- Validación

#####

```

#cargar archivos

load("Totalz30min26072023.RData")

load("Totalz1hr26072023.RData")

load("Totalz2hr26072023.RData")

load("Totalz6hr26072023.RData")

load("Totalz12hr26072023.RData")

load("Totalz24hr26072023.RData")

C30min = as.data.frame(totaldata_z30min)[,-c(1,22)]

C1hr = as.data.frame(totaldata_z1hr)[,-c(1,22)]

C2hr = as.data.frame(totaldata_z2hr)[,-c(1,22)]

C6hr = as.data.frame(totaldata_z6hr)[,-c(1,22)]

C12hr = as.data.frame(totaldata_z12hr)[,-c(1,22)]

C24hr = as.data.frame(totaldata_z24hr)[,-c(1,22)]

#####

#30 min

#####

#separar por institucion

df_imer30amin = C30min[,26:40]

df_imer30bmin = C30min[,21:25]

df_epmaps30min= C30min[,1:15]

df_inamhi30min = C30min[,16:20]

#calculo validacion

val30amin = NA

for (i in 1:ncol(df_imer30amin)) {

  imerg30amin = df_imer30amin[,i]

```

```

epmaps30min = df_epmaps30min[,i]

CC = cor(imerg30amin, epmaps30min, method = c("spearman"), use =
"pairwise.complete.obs")

RMSE <- rmse(imerg30amin, epmaps30min, na.rm = TRUE)

PBIAS <- pbias(imerg30amin, epmaps30min, na.rm = TRUE) #sim, obs

NSE <- NSE(imerg30amin, epmaps30min, na.rm = TRUE) #sim, obs

val3a = data.frame(CC, RMSE, PBIAS, NSE)

val30amin = rbind.data.frame(val30amin, val3a)
}

val30amin = val30amin[-1,]

rownames(val30amin) = colnames(df_epmaps30min)

val30bmin = NA

for (i in 1:ncol(df_imerg30bmin)) {

  imerg30bmin = df_imerg30bmin[,i]

  inamhi30min = df_inamhi30min[,i]

  CC = cor(imerg30bmin, inamhi30min, method = c("spearman"), use =
"pairwise.complete.obs")

  RMSE <- rmse(imerg30bmin, inamhi30min, na.rm = TRUE)

  PBIAS <- pbias(imerg30bmin, inamhi30min, na.rm = TRUE) #sim, obs

  NSE <- NSE(imerg30bmin, inamhi30min, na.rm = TRUE) #sim, obs

  val3b = data.frame(CC, RMSE, PBIAS, NSE)

  val30bmin = rbind.data.frame(val30bmin, val3b)

}

val30bmin = val30bmin[-1,]

rownames(val30bmin) = colnames(df_inamhi30min)

```

```
#####

#1h

#####

#separar por institucion

df_imer1ahr = C1hr[,26:40]
df_imer1bhr = C1hr[,21:25]
df_epmaps1hr= C1hr[,1:15]
df_inamhi1hr = C1hr[,16:20]

#calculo validacion

val1ahr = NA

for (i in 1:ncol(df_imer1ahr)) {

  imer1ahr = df_imer1ahr[,i]

  epmaps1hr = df_epmaps1hr[,i]

  CC = cor(imer1ahr, epmaps1hr, method = c("spearman"), use =
"pairwise.complete.obs")

  RMSE <- rmse(imer1ahr, epmaps1hr,na.rm = TRUE)

  PBIAS <- pbias(imer1ahr, epmaps1hr,na.rm = TRUE) #sim, obs

  NSE <- NSE(imer1ahr, epmaps1hr,na.rm = TRUE) #sim, obs

  val1a1 = data.frame(CC,RMSE,PBIAS,NSE)

  val1ahr = rbind.data.frame(val1ahr,val1a1)

}

val1ahr = val1ahr[-1,]

rownames(val1ahr) = colnames(df_epmaps1hr)

val1bhr = NA

for (i in 1:ncol(df_imer1bhr)) {

  imer1bhr = df_imer1bhr[,i]
```

```

inamhi1hr = df_inamhi1hr[,i]

CC = cor(imerg1bhr, inamhi1hr, method = c("spearman"), use = "pairwise.complete.obs")

RMSE <- rmse(imerg1bhr, inamhi1hr,na.rm = TRUE)

PBIAS <- pbias(imerg1bhr, inamhi1hr,na.rm = TRUE) #sim, obs

NSE <- NSE(imerg1bhr, inamhi1hr,na.rm = TRUE) #sim, obs

val1b1 = data.frame(CC,RMSE,PBIAS,NSE)

val1bhr = rbind.data.frame(val1bhr,val1b1)

}

val1bhr = val1bhr[-1,]

rownames(val1bhr) = colnames(df_inamhi1hr)

#####

#2h

#####

#separar por institucion

df_imerg2ahr = C2hr[,26:40]

df_imerg2bhr = C2hr[,21:25]

df_epmaps2hr= C2hr[,1:15]

df_inamhi2hr = C2hr[,16:20]

#calculo validacion

val2ahr = NA

for (i in 1:ncol(df_imerg2ahr)) {

  imerg2ahr = df_imerg2ahr[,i]

  epmaps2hr = df_epmaps2hr[,i]

  CC = cor(imerg2ahr, epmaps2hr, method = c("spearman"), use =
"pairwise.complete.obs")

  RMSE <- rmse(imerg2ahr, epmaps2hr,na.rm = TRUE)

```

```

PBIAS <- pbias(imerg2ahr, epmaps2hr,na.rm = TRUE) #sim, obs
NSE <- NSE(imerg2ahr, epmaps2hr,na.rm = TRUE) #sim, obs
val2a2 = data.frame(CC,RMSE,PBIAS,NSE)
val2ahr = rbind.data.frame(val2ahr,val2a2)
}
val2ahr = val2ahr[-1,]
rownames(val2ahr) = colnames(df_epmaps2hr)
val2bhr = NA
for (i in 1:ncol(df_imerg2bhr)) {
  imerg2bhr = df_imerg2bhr[,i]
  inamhi2hr = df_inamhi2hr[,i]
  CC = cor(imerg2bhr, inamhi2hr, method = c("spearman"), use = "pairwise.complete.obs")
  RMSE <- rmse(imerg2bhr, inamhi2hr,na.rm = TRUE)
  PBIAS <- pbias(imerg2bhr, inamhi2hr,na.rm = TRUE) #sim, obs
  NSE <- NSE(imerg2bhr, inamhi2hr,na.rm = TRUE) #sim, obs
  val2b2 = data.frame(CC,RMSE,PBIAS,NSE)
  val2bhr = rbind.data.frame(val2bhr,val2b2)
}
val2bhr = val2bhr[-1,]
rownames(val2bhr) = colnames(df_inamhi2hr)
#####
#6h
#####
#separar por institucion
df_imerg6ahr = C6hr[,26:40]

```

```

df_imerg6bhr = C6hr[,21:25]
df_epmaps6hr= C6hr[,1:15]
df_inamhi6hr = C6hr[,16:20]

#calculo validacion

val6ahr = NA

for (i in 1:ncol(df_imerg6ahr)) {

  imerg6ahr = df_imerg6ahr[,i]

  epmaps6hr = df_epmaps6hr[,i]

  CC = cor(imerg6ahr, epmaps6hr, method = c("spearman"), use =
"pairwise.complete.obs")

  RMSE <- rmse(imerg6ahr, epmaps6hr,na.rm = TRUE)

  PBIAS <- pbias(imerg6ahr, epmaps6hr,na.rm = TRUE) #sim, obs

  NSE <- NSE(imerg6ahr, epmaps6hr,na.rm = TRUE) #sim, obs

  val6a6 = data.frame(CC,RMSE,PBIAS,NSE)

  val6ahr = rbind.data.frame(val6ahr,val6a6)

}

val6ahr = val6ahr[-1,]

rownames(val6ahr) = colnames(df_epmaps6hr)

val6bhr = NA

for (i in 1:ncol(df_imerg6bhr)) {

  imerg6bhr = df_imerg6bhr[,i]

  inamhi6hr = df_inamhi6hr[,i]

  CC = cor(imerg6bhr, inamhi6hr, method = c("spearman"), use = "pairwise.complete.obs")

  RMSE <- rmse(imerg6bhr, inamhi6hr,na.rm = TRUE)

  PBIAS <- pbias(imerg6bhr, inamhi6hr,na.rm = TRUE) #sim, obs

  NSE <- NSE(imerg6bhr, inamhi6hr,na.rm = TRUE) #sim, obs

```



```

val6b6 = data.frame(CC, RMSE, PBIAS, NSE)

val6bhr = rbind.data.frame(val6bhr, val6b6)

}

val6bhr = val6bhr[-1,]

rownames(val6bhr) = colnames(df_inamhi6hr)

#####

#12h

#####

#separar por institucion

df_imerg12ahr = C12hr[,26:40]

df_imerg12bhr = C12hr[,21:25]

df_epmaps12hr = C12hr[,1:15]

df_inamhi12hr = C12hr[,16:20]

#calculo validacion

val12ahr = NA

for (i in 1:ncol(df_imerg12ahr)) {

  imerg12ahr = df_imerg12ahr[,i]

  epmaps12hr = df_epmaps12hr[,i]

  CC = cor(imerg12ahr, epmaps12hr, method = c("spearman"), use =
"pairwise.complete.obs")

  RMSE <- rmse(imerg12ahr, epmaps12hr, na.rm = TRUE)

  PBIAS <- pbias(imerg12ahr, epmaps12hr, na.rm = TRUE) #sim, obs

  NSE <- NSE(imerg12ahr, epmaps12hr, na.rm = TRUE) #sim, obs

  val12a12 = data.frame(CC, RMSE, PBIAS, NSE)

  val12ahr = rbind.data.frame(val12ahr, val12a12)

}

```

```

val12ahr = val12ahr[-1,]

rownames(val12ahr) = colnames(df_epmaps12hr)

val12bhr = NA

for (i in 1:ncol(df_imerg12bhr)) {

  imerg12bhr = df_imerg12bhr[,i]

  inamhi12hr = df_inamhi12hr[,i]

  CC = cor(imerg12bhr, inamhi12hr, method = c("spearman"), use =
"pairwise.complete.obs")

  RMSE <- rmse(imerg12bhr, inamhi12hr,na.rm = TRUE)

  PBIAS <- pbias(imerg12bhr, inamhi12hr,na.rm = TRUE) #sim, obs

  NSE <- NSE(imerg12bhr, inamhi12hr,na.rm = TRUE) #sim, obs

  val12b12 = data.frame(CC,RMSE,PBIAS,NSE)

  val12bhr = rbind.data.frame(val12bhr,val12b12)

}

val12bhr = val12bhr[-1,]

rownames(val12bhr) = colnames(df_inamhi12hr)

#####

#24h

#####

#separar por institucion

df_imerg24ahr = C24hr[,26:40]

df_imerg24bhr = C24hr[,21:25]

df_epmaps24hr= C24hr[,1:15]

df_inamhi24hr = C24hr[,16:20]

#calculo validacion

val24ahr = NA

```

```

for (i in 1:ncol(df_imerg24ahr)) {
  imerg24ahr = df_imerg24ahr[,i]
  epmaps24hr = df_epmaps24hr[,i]

  CC = cor(imerg24ahr, epmaps24hr, method = c("spearman"), use =
"pairwise.complete.obs")

  RMSE <- rmse(imerg24ahr, epmaps24hr,na.rm = TRUE)

  PBIAS <- pbias(imerg24ahr, epmaps24hr,na.rm = TRUE) #sim, obs

  NSE <- NSE(imerg24ahr, epmaps24hr,na.rm = TRUE) #sim, obs

  val24a24 = data.frame(CC,RMSE,PBIAS,NSE)

  val24ahr = rbind.data.frame(val24ahr,val24a24)
}

val24ahr = val24ahr[-1,]

rownames(val24ahr) = colnames(df_epmaps24hr)

val24bhr = NA

for (i in 1:ncol(df_imerg24bhr)) {
  imerg24bhr = df_imerg24bhr[,i]
  inamhi24hr = df_inamhi24hr[,i]

  CC = cor(imerg24bhr, inamhi24hr, method = c("spearman"), use =
"pairwise.complete.obs")

  RMSE <- rmse(imerg24bhr, inamhi24hr,na.rm = TRUE)

  PBIAS <- pbias(imerg24bhr, inamhi24hr,na.rm = TRUE) #sim, obs

  NSE <- NSE(imerg24bhr, inamhi24hr,na.rm = TRUE) #sim, obs

  val24b24 = data.frame(CC,RMSE,PBIAS,NSE)

  val24bhr = rbind.data.frame(val24bhr,val24b24)
}

val24bhr = val24bhr[-1,]

```

```

rownames(val24bhr) = colnames(df_inamhi24hr)

#####

#juntar los resultados de validacion

val0.5h = rbind.data.frame(val30amin,val30bmin)

val0.5h$duracion = rep("30min",nrow(val0.5h))

val1h = rbind.data.frame(val1ahr,val1bhr)

val1h$duracion = rep("1h",nrow(val1h))

val2h = rbind.data.frame(val2ahr,val2bhr)

val2h$duracion = rep("2h",nrow(val2h))

val6h = rbind.data.frame(val6ahr,val6bhr)

val6h$duracion = rep("6h",nrow(val6h))

val12h = rbind.data.frame(val12ahr,val12bhr)

val12h$duracion = rep("12h",nrow(val12h))

val24h = rbind.data.frame(val24ahr,val24bhr)

val24h$duracion = rep("24h",nrow(val24h))

val_tot = rbind.data.frame(val0.5h,val1h,val2h,val6h,val12h,val24h)

val_tot$estacion = rownames(val_tot)

val_tot$estacion2=
rep(c("C02","C04","C05","C14","C15","P03","P08","P09","P11","P12","P27","P28","P31","P
32","P70",

      "M0003","M0006","M0007","M0025","M0033"),6)

##plots bar

val_tot$duracion =factor(val_tot$duracion, levels= c("30min","1h","2h","6h","12h","24h"))

plt_CC = ggplot(val_tot, aes(x=estacion2, y=CC, fill=duracion))+

  geom_bar(position="dodge", stat="identity")+ theme_bw()+

  xlab("ESTACIÓN") + ylab("COEFICIENTE PEARSON [-]")+

```

```

scale_fill_manual(values =
c("#FFCDD2", "#EF9A9A", "#EF5350", "#F44336", "#D32F2F", "#B71C1C"))

plt_RMSE = ggplot(val_tot, aes(x=estacion2, y=RMSE, fill=duracion))+

geom_bar(position="dodge", stat="identity")+ theme_bw()+

xlab("ESTACIÓN") + ylab("RMSE [mm]")+

scale_fill_manual(values =
c("#FFCDD2", "#EF9A9A", "#EF5350", "#F44336", "#D32F2F", "#B71C1C"))

plt_PBIAS = ggplot(val_tot, aes(x=estacion2, y=PBIAS, fill=duracion))+

geom_bar(position="dodge", stat="identity")+ theme_bw()+

xlab("ESTACIÓN") + ylab("PBIAS [%]")+

scale_fill_manual(values =
c("#FFCDD2", "#EF9A9A", "#EF5350", "#F44336", "#D32F2F", "#B71C1C"))

plt_NSE = ggplot(val_tot, aes(x=estacion2, y=NSE, fill=duracion))+

geom_bar(position="dodge", stat="identity")+ theme_bw()+

xlab("ESTACIÓN") + ylab("NSE [-]")+

scale_fill_manual(values =
c("#FFCDD2", "#EF9A9A", "#EF5350", "#F44336", "#D32F2F", "#B71C1C"))

plt_fig3 <- ggarrange(plt_CC, plt_RMSE, plt_PBIAS, plt_NSE,

common.legend = T, legend = "bottom",

ncol = 2, nrow = 2, align = "v")

png(file='fig10_val_plots.png', width = 1000, height = 600)

plt_fig3

dev.off()

#####

Anexo 6. Análisis CDF para la serie de 1hr

#----- TIC SPP IDF -----

```

```
#----- Análisis CDF -----
```

```
#LIBRERIAS
```

```
library(conflicted)
```

```
library(tidyverse)
```

```
library(lubridate)
```

```
library(dplyr)
```

```
library(fitdistrplus)
```

```
library(PearsonDS)
```

```
library(caret)
```

```
library(lubridate)
```

```
library(openxlsx)
```

```
library(minpack.lm)
```

```
library(investr)
```

```
library(dplyr)
```

```
library(xts)
```

```
library(sp)
```

```
library(rgdal)
```

```
library(Metrics)
```

```
library(hydroGOF)
```

```
library(zoo)
```

```
#Borrar variables
```

```
rm(list = ls())
```

```
#CARGAR ARCHIVOS
```

```
load("Totalz1hr26072023.RData")
```

```
C1hr = as.data.frame(totaldata_z1hr)[-c(1,22)]
```

```

ind1hr = as.POSIXct(rownames(C1hr),format='%Y-%m-%d %H:%M:%S')

#####

#1.- Bucle principal

#####

iteraciones = NA

resultadosAMS <- zoo(NA, as.POSIXct(paste(2000:2023,"01", "01", sep = "-"), format=
"%Y-%m-%d"))

resultadosCHI <- NA

resultadosKS <- NA

resultadosMag <- NA

resultadosRES <- NA

resultadosGOF <- NA

#correr por todas las columnas de C2

for (j in 1:40) {

  ams_1hr <- data.frame(ind1hr, C1hr[, j])

  colnames(ams_1hr) <- c("Fecha", "pmax")

  print(paste(j, "de 40"))

  #Obtener la serie maxima anual de cada estación

  ams_1hr <- ams_1hr %>%

    group_by(Fecha = floor_date(Fecha, "year")) %>%

    summarise(pmax = max(pmax, na.rm = T))

  ams_1hr$Fecha <- year(ams_1hr$Fecha)

  #Cambiarlas filas que contengan algun valor nulo a NA

  ams_1hr[ams_1hr == -Inf] <- NA

  ams_1hr[ams_1hr == 0] <- NA

  #Eliminar todas las filas que contengan algun valor NA

```

```

ams_1hr <- na.omit(ams_1hr)

#Convertir en zoo

z_ams_1hr = zoo(ams_1hr$pmax,as.POSIXct(paste(ams_1hr$Fecha,"01","01",sep = "-
"), format= "%Y-%m-%d"))

resultadosAMS = merge(resultadosAMS,z_ams_1hr)

#plot(ams_1hr, type = "h", xlab = "Año", ylab = "Precipitación máxima diaria [mm]")

if (nrow(ams_1hr)<3) {

  print(paste("menor que 3 =", j, sep=" "))

  next

}

#iteraciones efectivas

iteraciones = c(iteraciones, j)

#Distribución Gumbel

#p: función de distribución

dgumbel <- function(x, a, b) 1/b*exp((a-x)/b)*exp(-exp((a-x)/b))

pgumbel <- function(q, a, b) exp(-exp((a-q)/b))

qgumbel <- function(p, a, b) a-b*log(-log(p))

Gumbel<-fitdist(ams_1hr$pmax`, "gumbel", start=list(a=10, b=10))

#plot(pgumbel(0:10,10,0.6),type="h",xlab="k",ylab="P(X=k)",main="Función de
Probabilidad de Gumbel")

#Distribución Pearson III

#Se definen las funciones de densidad, distribución y cuantiles

dPIII <- function(x, shape, location, scale) dpearsonIII(x, shape, location, scale,
log=FALSE)

pPIII <- function(q, shape, location, scale) ppearsonIII(q, shape, location, scale, lower.tail
= TRUE, log.p = FALSE)

```



```

qPIII <- function(p, shape, location, scale) qpearsonIII(p, shape, location, scale, lower.tail
= TRUE, log.p = FALSE)

#Funciones generadoras de momentos (Para ajuste por método de los momentos)

mPIII <- function(order, shape, location, scale) # compute the empirical first 3 moments
of the PIII distribution

{
  if(order == 1) return(location + shape * scale)

  if(order == 2) return(scale * scale * shape)

  if(order == 3) return(2 / sqrt(shape) * sign(scale))
}

ePIII<-function(x, order) # compute (centered) empirical centered moments of the data
{
  if(order == 1) return(mean(x))

  if(order == 2) return(var(x))

  if(order == 3) return(e1071::skewness(x, type = 2))
}

#Estadígrafos para Pearson III

m <- mean(ams_1hr$pmax)
v <- var(ams_1hr$pmax)
s <- sd(ams_1hr$pmax)
g <- e1071::skewness(ams_1hr$pmax, type=2)

#Se generan los parámetros iniciales

# Correct the sample skew for bias using the recommendation of

# Bobee, B. and R. Robitaille (1977). "The use of the Pearson Type 3 and Log Pearson
Type 3 distributions revisited."

# Water Resources Reseach 13(2): 427-443, as used by Kite

```

```

my_shape<-(2 / g) ^ 2

my_scale<-sqrt(v / my_shape) * sign(g) # modified as recommended by Carl Schwarz

my_location<-m-my_scale*my_shape

start<-list(shape = my_shape, location = my_location, scale = my_scale)

#Finalmente, se ajusta la distribución Pearson III

PIII<-fitdist(ams_1hr$pmax, "PIII", start = start, method = "mme", order = 1:3, memp =
ePIII, control = list(maxit = 1000))

#Distribución Log Pearson III

#El proceso es idéntico al seguido para la distribución Pearson III. Sólo se utiliza el
logaritmo de los datos.

dPIII <- function(x, shape, location, scale) dpearsonIII(log(x), shape, location, scale,
log=FALSE)/x

pPIII <- function(q, shape, location, scale) ppearsonIII(log(q), shape, location, scale,
lower.tail = TRUE, log.p = FALSE)

qPIII <- function(p, shape, location, scale) exp(qpearsonIII(p, shape, location, scale,
lower.tail = TRUE, log.p = FALSE))

#Funciones generadores de momentos (logarítmicas):

mPIII <- function(order, shape, location, scale)

{

# compute the empirical first 3 moments of the PIII distribution

if(order == 1) return(location + shape * scale)

if(order == 2) return(scale * scale * shape)

if(order == 3) return(2 / sqrt(shape) * sign(scale))

}

ePIII <- function(x, order)

{

```

```

# compute (centered) empirical centered moments of the data

if(order == 1) return(mean(log(x)))

if(order == 2) return(var(log(x)))

if(order == 3) return(e1071::skewness(log(x), type = 2))

}

#Estadígrafos para Log Pearson III

m <- mean(log(ams_1hr$pmax))

v <- var(log(ams_1hr$pmax))

s <- sd(log(ams_1hr$pmax))

g <- e1071::skewness(log(ams_1hr$pmax), type=2)

#Se generan los parámetros iniciales para la estimación

# Correct the sample skew for bias using the recommendation of

# Bobee, B. and R. Robitaille (1977). "The use of the Pearson Type 3 and Log Pearson
Type 3 distributions revisited."

# Water Resources Reseach 13(2): 427-443, as used by Kite

my_shape <- (2 / g) ^ 2

my_scale <- sqrt(v / my_shape) * sign(g)

my_location <- m - my_scale * my_shape

start <- list(shape = my_shape, location = my_location, scale = my_scale)

#Finalmente, se ajusta la distribución Log Pearson III

LP3 <- fitdist(ams_1hr$pmax, "LP3", start = start, method = "mme", order = 1:3, memp
= eLP3, control = list(maxit = 1000))

#Distribucion Log Normal

LogNormal <- fitdist(ams_1hr$pmax, distr = "lnorm", method = "mme")

#Probabilidades de ocurrencia teóricas

```

#A continuación, se agregan las probabilidades de ocurrencia teóricas de las distribuciones ajustadas.

```
Poc <- data.frame(
  "P" = sort(ams_1hr$pmax, decreasing = TRUE),
  "Pexc" = (1:nrow(ams_1hr))/(nrow(ams_1hr)+1)
)

Poc$Pocu <- 1 - Poc$Pexc

Poc$LogNormal <- pnorm(log(Poc$P), LogNormal$estimate[1], LogNormal$estimate[2])
Poc$Gumbel <- pgumbel(Poc$P, Gumbel$estimate[1], Gumbel$estimate[2])
Poc$PIII <- pPIII(Poc$P, PIII$estimate[1], PIII$estimate[2], PIII$estimate[3])
Poc$LPIII <- pLPIII(Poc$P, LPIII$estimate[1], LPIII$estimate[2], LPIII$estimate[3])

#head(Poc)

#GRAFICOS

#Evaluación de ajustes

#Gráficos de densidad

#denscomp(list(Gumbel, LogNormal, PIII, LPIII), addlegend = TRUE,
#legendtext = c("Gumbel", "LogNormal", "Pearson III", "LogPearson III"))

#Gráficos de densidad acumulada

#cdfcomp(list( Gumbel, LogNormal, PIII, LPIII), addlegend = TRUE,
#legendtext = c( "Gumbel", "LogNormal", "Pearson III", "LogPearson III"))

#par(mfrow=c(3,2))

#cdfcomp(LogNormal, addlegend = FALSE, main = "Log Normal")
#cdfcomp(Gumbel, addlegend = FALSE, main = "Gumbel")
#cdfcomp(PIII, addlegend = FALSE, main = "Pearson III")
#cdfcomp(LPIII, addlegend = FALSE, main = "Log Pearson III")

#Pruebas de bondad de ajuste
```

#Mediante el comando `gofstat()` del paquete `fitdistrplus` se pueden ejecutar diversas pruebas de bondad de ajuste: Kolmogorov - Smirnov, Cramer - von Misses y Anderson - Darling.

```
if (nrow(ams_1hr)>10) {  
  GOF <- gofstat(list(LogNormal,Gumbel, PIII, LPIII),  
                 fitnames = c("LogNormal", "Gumbel", "Pearson III", "LogPearson III"))  
  
  #print(GOF)  
  
  GOF$chisq  
  
  #knitr::kable(GOF$chisqpvalue)  
  
  GOF$names = rep(colnames(C1hr)[j],4)  
  
  #Guardar resultados  
  
  resultadosCHI= cbind.data.frame(resultadosCHI,GOF$chisqpvalue,GOF$names)  
}  
  
kslnorm <- ks.test(Poc$Pocu, Poc$LogNormal)  
ksgumbel <- ks.test(Poc$Pocu, Poc$Gumbel)  
kspiii <- ks.test(Poc$Pocu, Poc$PIII)  
kslpiii <- ks.test(Poc$Pocu, Poc$LPIII)  
  
KS <- data.frame(c(  
  kslnorm$statistic,  
  ksgumbel$statistic,  
  kspiii$statistic,  
  kslpiii$statistic),  
  c(kslnorm$p.value,  
    ksgumbel$p.value,  
    kspiii$p.value,
```

```

kslpiii$p.value))

colnames(KS) <- c("D", "P-Value")

rownames(KS) <- c("LogNormal", "Gumbel", "Pearson III", "LogPearson III")

KS <- round(KS, 3)

KS$names = rep(colnames(C1hr)[j],4)

#head(KS)

#Guardar resultados

resultadosKS= rbind.data.frame(resultadosKS,KS)

#GRAFICO

#matplot(Magnitudes[,3:6], type = c("l"), col = 1:4, ylab = "Precipitación Máxima Diaria
[mm]", xlab = "T [años]")

#legend("bottomright",

#legend = c("LogNormal", "Gumbel", "Pearson III", "LogPearson III"),

#col=1:4, pch=15, cex = 0.8)

T <- c(2,5,10,50,100)

Resumen <- data.frame("T" = T,

                      exp(qnorm(1 - 1/T, LogNormal$estimate[1], LogNormal$estimate[2])),

                      qgumbel(1 - 1/T, Gumbel$estimate[1], Gumbel$estimate[2]),

                      qPIII(1 - 1/T, PIII$estimate[1], PIII$estimate[2], PIII$estimate[3]),

                      qlPIII(1 - 1/T, LPIII$estimate[1], LPIII$estimate[2], LPIII$estimate[3]))

colnames(Resumen) <- c("T", "LogNormal", "Gumbel", "Pearson III", "LogPearson III")

Resumen$names = rep(colnames(C1hr)[j],5)

#head(KS)

#Guardar resultados

resultadosRES = rbind.data.frame(resultadosRES,Resumen)

```

```

knitr::kable(Resumen)
}
iteraciones = iteraciones[-1]
resultadosAMS_1 = as.data.frame(resultadosAMS)[-1]
resultadosAMS_1 = data.frame(index(resultadosAMS),resultadosAMS_1)
colnames(resultadosAMS_1) <- c("TS",colnames(C1hr))
resultadosCHI = resultadosCHI[-1]
resultadosKS = resultadosKS[-1,]
#SACAR RESULTADOS EN UN EXCEL
write.csv(resultadosCHI,file="TestCHI1hr.csv")
write.csv(resultadosKS,file="TestKS1hr.csv")
write.csv(resultadosAMS_1,file="AMS1hr.csv")
write.csv(resultadosRES,file="RESUMEN1hr.csv")

```

```
#-----
```

Anexo 7. Script gráfico de curvas IDF

```
#----- TIC SPP IDFs -----
```

```
#----- GRAFICO IDFs -----
```

```

library(lmomco)
library(lubridate)
library(openxlsx)
library(minpack.lm)
library(investr)
library(dplyr)
library(xts)
library(sp)

```

```

library(rgdal)

library(Metrics)

library(hydroGOF)

library(zoo)

library(reshape)

library(ggplot2)

library(ggpubr)

rm(list=ls())

Dur = c(0.5,1,2,6,12,24)

yhlabC15 = c(20,40,60,80,100)

yhlabM03 = c(10,20,30,40,50)

yhlabM07 = c(50,100,150,200,250)

yhlabM25= c(10,20,30,40,50,60)

yhlabM33= c(20,40,60,80)

yhlabM06 = c(20,40,60,80,100,120)

yhlab25 = c(25,50,75,100,125,150,175)

#####

#1.- CARGAR ARCHIVOS

#####

Dur = c(0.5,1,2,6,12,24)

yhlabC15 = c(20,40,60,80,100)

#C15

C15O = read.csv(file="C15EPMAPSNANEGALITO.csv")

C15I = read.csv(file="C15IMERG.csv")

C15O = C15O[-c(1:25,28,29,31),]

```



```

C15I = C15I[-c(1:25,28,29,31),]
C15O = C15O[-c(1)]
C15I = C15I[-c(1)]
colnames(C15O) = c("0.5hr","1hr","2hr","6hr","12hr","24hr")
colnames(C15I) = c("0.5hr","1hr","2hr","6hr","12hr","24hr")
C15O[1,] = c("0.5","1","2","6","12","24")
C15I[1,] = c("0.5","1","2","6","12","24")
C15O[,1] = as.numeric(C15O[,1])/0.5 #de pp[mm] a intensidad [mm/h]
C15O[,2] = as.numeric(C15O[,2])/1 #de pp[mm] a intensidad [mm/h]
C15O[,3] = as.numeric(C15O[,3])/2 #de pp[mm] a intensidad [mm/h]
C15O[,4] = as.numeric(C15O[,4])/6 #de pp[mm] a intensidad [mm/h]
C15O[,5] = as.numeric(C15O[,5])/12 #de pp[mm] a intensidad [mm/h]
C15O[,6] = as.numeric(C15O[,6])/24 #de pp[mm] a intensidad [mm/h]
C15I[,1] = as.numeric(C15I[,1])/0.5 #de pp[mm] a intensidad [mm/h]
C15I[,2] = as.numeric(C15I[,2])/1 #de pp[mm] a intensidad [mm/h]
C15I[,3] = as.numeric(C15I[,3])/2 #de pp[mm] a intensidad [mm/h]
C15I[,4] = as.numeric(C15I[,4])/6 #de pp[mm] a intensidad [mm/h]
C15I[,5] = as.numeric(C15I[,5])/12 #de pp[mm] a intensidad [mm/h]
C15I[,6] = as.numeric(C15I[,6])/24 #de pp[mm] a intensidad [mm/h]
C15O[1,1] = C15O[1,1]*0.5
C15O[1,2] = C15O[1,2]*1
C15O[1,3] = C15O[1,3]*2
C15O[1,4] = C15O[1,4]*6
C15O[1,5] = C15O[1,5]*12
C15O[1,6] = C15O[1,6]*24

```

```

C15l[1,1] = C15l[1,1]*0.5
C15l[1,2] = C15l[1,2]*1
C15l[1,3] = C15l[1,3]*2
C15l[1,4] = C15l[1,4]*6
C15l[1,5] = C15l[1,5]*12
C15l[1,6] = C15l[1,6]*24
C15O_2 = C15O[-c(3),]
C15O_50 = C15O[-c(2),]
C15O_2 = t(C15O_2)
colnames(C15O_2) = c("d","i")
C15O_50 = t(C15O_50)
colnames(C15O_50) = c("d","i")
C15l_2 = C15l[-c(3),]
C15l_50 = C15l[-c(2),]
C15l_2 = t(C15l_2)
colnames(C15l_2) = c("d","i")
C15l_50 = t(C15l_50)
colnames(C15l_50) = c("d","i")
png("idfc15.png",width = 700, height = 400)
plt_C15 = {plot(C15O_2, type="o", col = "red", lwd=1,font = 2,main = "C15",
  ylab="", lty=2, xlab= "", cex.lab=1.1, cex.main=2,yaxt = "n",
  xaxt = "n",xlim = c(0, 25), ylim = c(0, 90))
lines(C15l_2, type="o", col = "blue", lty=1 , lwd=1)
lines(C15O_50, type="o", col = "red3", lty=2, lwd=2 )
lines(C15l_50, type="o", col = "blue4", lty=1 , lwd=2)

```

```

title(xlab= "Duración (h)", line=2.3, cex.lab=1.1)
title(ylab= "Intensidad (mm/h)", line=2.3, cex.lab=1.1)
abline(v=Dur, col="snow3", lwd=0.5)
abline(h=yhlabC15, col="snow3", lwd=0.5)
label <- c("Tr 2 years OBS", "Tr 2 years IMERG", "Tr 50 years OBS", "Tr 50 years IMERG")
legend("topright", label, col = c("red", "blue", "red3", "blue4") ,
      lty=c(2,1,2,1), pch = 1, lwd=1, y.intersp=1, cex = 1 )
axis(1, at = c(0.5,1,2,6,12,24),
      labels = c("0.5", "1", "2", "6", "12", "24"))
axis(2, at = c(20,40,60,80,100),
      labels = c("20", "40", "60", "80", "100"))
dev.off()

#-----
#M03
M03O = read.csv(file="M03INAMHI.csv")
M03I = read.csv(file="M03IMERG.csv")
M03O = M03O[-c(1:25,28,29,31),]
M03I = M03I[-c(1:25,28,29,31),]
M03O = M03O[,-c(1)]
M03I = M03I[,-c(1)]
colnames(M03O) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")
colnames(M03I) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")
M03O[1,] = c("0.5", "1", "2", "6", "12", "24")
M03I[1,] = c("0.5", "1", "2", "6", "12", "24")

```

```

M03I[,1] = as.numeric(M03I[,1])/0.5 #de pp[mm] a intensidad [mm/h]
M03I[,2] = as.numeric(M03I[,2])/1 #de pp[mm] a intensidad [mm/h]
M03I[,3] = as.numeric(M03I[,3])/2 #de pp[mm] a intensidad [mm/h]
M03I[,4] = as.numeric(M03I[,4])/6 #de pp[mm] a intensidad [mm/h]
M03I[,5] = as.numeric(M03I[,5])/12 #de pp[mm] a intensidad [mm/h]
M03I[,6] = as.numeric(M03I[,6])/24 #de pp[mm] a intensidad [mm/h]

M03I[1,1] = M03I[1,1]*0.5
M03I[1,2] = M03I[1,2]*1
M03I[1,3] = M03I[1,3]*2
M03I[1,4] = M03I[1,4]*6
M03I[1,5] = M03I[1,5]*12
M03I[1,6] = M03I[1,6]*24

M03O_2 = M03O[-c(3),]
M03O_50 = M03O[-c(2),]
M03O_2 = t(M03O_2)
colnames(M03O_2) = c("d","i")
M03O_50 = t(M03O_50)
colnames(M03O_50) = c("d","i")

M03I_2 = M03I[-c(3),]
M03I_50 = M03I[-c(2),]
M03I_2 = t(M03I_2)
colnames(M03I_2) = c("d","i")
M03I_50 = t(M03I_50)
colnames(M03I_50) = c("d","i")

```

```

png("idfM03.png",width = 700, height = 400)

plt_M03 = {plot(M03O_2, type="o", col = "red", lwd=1,font = 2,main = "M0003",
              ylab="", lty=2, xlab= "", cex.lab=2, cex.main=2,yaxt = "n",
              xaxt = "n",xlim = c(0, 25), ylim = c(0, 70))

lines(M03I_2, type="o", col = "blue", lty=1 , lwd=1)

lines(M03O_50, type="o", col = "red3", lty=2, lwd=2 )

lines(M03I_50, type="o", col ="blue4", lty=1 , lwd=2)

title(xlab= "Duración (h)", line=2.3, cex.lab=1.1)

title(ylab= "Intensidad (mm/h)", line=2.3, cex.lab=1.1)

abline(v=Dur, col="snow3", lwd=0.5)

abline(h=yhlabM33, col="snow3", lwd=0.5)

label <- c("Tr 2 years OBS", "Tr 2 years IMERG", "Tr 50 years OBS", "Tr 50 years
IMERG")

legend("topright", label, col = c("red", "blue", "red3", "blue4") ,
      lty=c(2,1,2,1), pch = 1, lwd=1, y.intersp=1, cex = 1 )

axis(1, at = c(0.5,1,2,6,12,24),
     labels = c("0.5", "1", "2", "6", "12", "24"))

axis(2, at = c(20,40,60,80),
     labels = c("20", "40", "60", "80"))}

dev.off()

#-----

#M06

M06O = read.csv(file="M06INAMHI .csv")

M06I = read.csv(file="M06IMERG.csv")

M06O = M06O[-c(1:25,28,29,31),]

M06I = M06I[-c(1:25,28,29,31),]

```

```

M06O = M06O[-c(1)]

M06I = M06I[-c(1)]

colnames(M06O) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")

colnames(M06I) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")

M06O[1,] = c("0.5", "1", "2", "6", "12", "24")

M06I[1,] = c("0.5", "1", "2", "6", "12", "24")

M06I[,1] = as.numeric(M06I[,1])/0.5 #de pp[mm] a intensidad [mm/h]

M06I[,2] = as.numeric(M06I[,2])/1 #de pp[mm] a intensidad [mm/h]

M06I[,3] = as.numeric(M06I[,3])/2 #de pp[mm] a intensidad [mm/h]

M06I[,4] = as.numeric(M06I[,4])/6 #de pp[mm] a intensidad [mm/h]

M06I[,5] = as.numeric(M06I[,5])/12 #de pp[mm] a intensidad [mm/h]

M06I[,6] = as.numeric(M06I[,6])/24 #de pp[mm] a intensidad [mm/h]

M06I[1,1] = M06I[1,1]*0.5

M06I[1,2] = M06I[1,2]*1

M06I[1,3] = M06I[1,3]*2

M06I[1,4] = M06I[1,4]*6

M06I[1,5] = M06I[1,5]*12

M06I[1,6] = M06I[1,6]*24

M06O_2 = M06O[-c(3),]

M06O_50 = M06O[-c(2),]

M06O_2 = t(M06O_2)

colnames(M06O_2) = c("d", "i")

M06O_50 = t(M06O_50)

colnames(M06O_50) = c("d", "i")

M06I_2 = M06I[-c(3),]

```

```

M06I_50 = M06I[-c(2),]
M06I_2 = t(M06I_2)
colnames(M06I_2) = c("d","i")
M06I_50 = t(M06I_50)
colnames(M06I_50) = c("d","i")
png("idfM06.png",width = 700, height = 400)
plt_M06 = {plot(M06O_2, type="o", col = "red", lwd=1,font = 2,main = "M0006",
               ylab="", lty=2, xlab= "", cex.lab=2 , cex.main=2 ,yaxt = "n",
               xaxt = "n",xlim = c(0, 25), ylim = c(0, 125))
lines(M06I_2, type="o", col = "blue", lty=1 , lwd=1)
lines(M06O_50, type="o", col = "red3", lty=2, lwd=2 )
lines(M06I_50, type="o", col ="blue4", lty=1 , lwd=2)
title(xlab= "Duración (h)", line=2.3, cex.lab=1.1)
title(ylab= "Intensidad (mm/h)", line=2.3, cex.lab=1.1)
abline(v=Dur, col="snow3", lwd=0.5)
abline(h=yhlabM06, col="snow3", lwd=0.5)
label <- c("Tr 2 years OBS","Tr 2 years IMERG","Tr 50 years OBS", "Tr 50 years
IMERG")
legend("topright", label, col = c("red","blue","red3","blue4") ,
       lty=c(2,1,2,1), pch = 1, lwd=1, y.intersp=1, cex = 1 )
axis(1, at = c(0.5,1,2,6,12,24),
      labels = c("0.5", "1", "2", "6", "12", "24"))
axis(2, at = c(20,40,60,80,100,120),
      labels = c("20", "40", "60", "80", "100", "120"))
dev.off()
#-----

```

```

#M07

M07O = read.csv(file="M07INAMHI .csv")

M07I = read.csv(file="M07IMERG.csv")

M07O = M07O[-c(1:25,28,29,31),]

M07I = M07I[-c(1:25,28,29,31),]

M07O = M07O[,-c(1)]

M07I = M07I[,-c(1)]

colnames(M07O) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")

colnames(M07I) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")

M07O[1,] = c("0.5", "1", "2", "6", "12", "24")

M07I[1,] = c("0.5", "1", "2", "6", "12", "24")

M07I[,1] = as.numeric(M07I[,1])/0.5 #de pp[mm] a intensidad [mm/h]

M07I[,2] = as.numeric(M07I[,2])/1 #de pp[mm] a intensidad [mm/h]

M07I[,3] = as.numeric(M07I[,3])/2 #de pp[mm] a intensidad [mm/h]

M07I[,4] = as.numeric(M07I[,4])/6 #de pp[mm] a intensidad [mm/h]

M07I[,5] = as.numeric(M07I[,5])/12 #de pp[mm] a intensidad [mm/h]

M07I[,6] = as.numeric(M07I[,6])/24 #de pp[mm] a intensidad [mm/h]

M07I[1,1] = M07I[1,1]*0.5

M07I[1,2] = M07I[1,2]*1

M07I[1,3] = M07I[1,3]*2

M07I[1,4] = M07I[1,4]*6

M07I[1,5] = M07I[1,5]*12

M07I[1,6] = M07I[1,6]*24

M07O_2 = M07O[-c(3),]

M07O_50 = M07O[-c(2),]

```



```

M07O_2 = t(M07O_2)
colnames(M07O_2) = c("d","i")
M07O_50 = t(M07O_50)
colnames(M07O_50) = c("d","i")
M07I_2 = M07I[-c(3),]
M07I_50 = M07I[-c(2),]
M07I_2 = t(M07I_2)
colnames(M07I_2) = c("d","i")
M07I_50 = t(M07I_50)
colnames(M07I_50) = c("d","i")
png("idfM07.png",width = 700, height = 400)
plt_M07 = {plot(M07O_2, type="o", col = "red", lwd=1,font = 2,main = "M0007",
               ylab="", lty=2, xlab= "", cex.lab=2 , cex.main=2 ,yaxt = "n",
               xaxt = "n",xlim = c(0, 25), ylim = c(0, 120))
lines(M07I_2, type="o", col = "blue", lty=1 , lwd=1)
lines(M07O_50, type="o", col = "red3", lty=2, lwd=2 )
lines(M07I_50, type="o", col = "blue4", lty=1 , lwd=2)
title(xlab= "Duración (h)", line=2.3, cex.lab=1.1)
title(ylab= "Intensidad (mm/h)", line=2.3, cex.lab=1.1)
abline(v=Dur, col="snow3", lwd=0.5)
abline(h=yhlabC15, col="snow3", lwd=0.5)
label <- c("Tr 2 years OBS", "Tr 2 years IMERG", "Tr 50 years OBS", "Tr 50 years
IMERG")
legend("topright", label, col = c("red", "blue", "red3", "blue4") ,
       lty=c(2,1,2,1), pch = 1, lwd=1, y.intersp=1, cex = 1 )
axis(1, at = c(0.5,1,2,6,12,24),

```

```

labels = c("0.5", "1", "2", "6", "12", "24"))
axis(2, at = c(20,40,60,80,100),
labels = c("20", "40", "60", "80", "100"))}
dev.off()

#-----

#M25

M25O = read.csv(file="M25INAMHI .csv")
M25I = read.csv(file="M25IMERG.csv")
M25O = M25O[-c(1:25,28,29,31),]
M25I = M25I[-c(1:25,28,29,31),]
M25O = M25O[,-c(1)]
M25I = M25I[,-c(1)]

colnames(M25O) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")
colnames(M25I) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")
M25O[1,] = c("0.5", "1", "2", "6", "12", "24")
M25I[1,] = c("0.5", "1", "2", "6", "12", "24")

M25I[,1] = as.numeric(M25I[,1])/0.5 #de pp[mm] a intensidad [mm/h]
M25I[,2] = as.numeric(M25I[,2])/1 #de pp[mm] a intensidad [mm/h]
M25I[,3] = as.numeric(M25I[,3])/2 #de pp[mm] a intensidad [mm/h]
M25I[,4] = as.numeric(M25I[,4])/6 #de pp[mm] a intensidad [mm/h]
M25I[,5] = as.numeric(M25I[,5])/12 #de pp[mm] a intensidad [mm/h]
M25I[,6] = as.numeric(M25I[,6])/24 #de pp[mm] a intensidad [mm/h]

M25I[1,1] = M25I[1,1]*0.5
M25I[1,2] = M25I[1,2]*1
M25I[1,3] = M25I[1,3]*2

```

```

M25I[1,4] = M25I[1,4]*6
M25I[1,5] = M25I[1,5]*12
M25I[1,6] = M25I[1,6]*24
M25O_2 = M25O[-c(3),]
M25O_50 = M25O[-c(2),]
M25O_2 = t(M25O_2)
colnames(M25O_2) = c("d","i")
M25O_50 = t(M25O_50)
colnames(M25O_50) = c("d","i")
M25I_2 = M25I[-c(3),]
M25I_50 = M25I[-c(2),]
M25I_2 = t(M25I_2)
colnames(M25I_2) = c("d","i")
M25I_50 = t(M25I_50)
colnames(M25I_50) = c("d","i")
png("idfM25.png",width = 700, height = 400)
plt_M25 = {plot(M25O_2, type="o", col = "red", lwd=1,font = 2,main = "M0025",
               ylab="", lty=2, xlab= "", cex.lab=2 , cex.main=2 ,yaxt = "n",
               yaxt = "n",xlim = c(0, 25), ylim = c(0,175))
lines(M25I_2, type="o", col = "blue", lty=1 , lwd=1)
lines(M25O_50, type="o", col = "red3", lty=2, lwd=2 )
lines(M25I_50, type="o", col ="blue4", lty=1 , lwd=2)
title(xlab= "Duración (h)", line=2.3, cex.lab=1.1)
title(ylab= "Intensidad (mm/h)", line=2.3, cex.lab=1.1)
abline(v=Dur, col="snow3", lwd=0.5)

```

```

abline(h=yhlab25, col="snow3", lwd=0.5)

label <- c("Tr 2 years OBS", "Tr 2 years IMERG", "Tr 50 years OBS", "Tr 50 years
IMERG")

legend("topright", label, col = c("red", "blue", "red3", "blue4") ,
      lty=c(2,1,2,1), pch = 1, lwd=1, y.intersp=1, cex = 1 )

axis(1, at = c(0.5,1,2,6,12,24),
     labels = c("0.5", "1", "2", "6", "12", "24"))

axis(2, at = c(25,50,75,100,125,150,175),
     labels = c("25", "50", "75", "100", "125", "150", "175"))

dev.off()

#-----

#M33

M33O = read.csv(file="M33INAMHI .csv")

M33I = read.csv(file="M33IMERG.csv")

M33O = M33O[-c(1:25,28,29,31),]

M33I = M33I[-c(1:25,28,29,31),]

M33O = M33O[,-c(1)]

M33I = M33I[,-c(1)]

colnames(M33O) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")

colnames(M33I) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")

M33O[1,] = c("0.5", "1", "2", "6", "12", "24")

M33I[1,] = c("0.5", "1", "2", "6", "12", "24")

M33I[,1] = as.numeric(M33I[,1])/0.5 #de pp[mm] a intensidad [mm/h]

M33I[,2] = as.numeric(M33I[,2])/1 #de pp[mm] a intensidad [mm/h]

M33I[,3] = as.numeric(M33I[,3])/2 #de pp[mm] a intensidad [mm/h]

M33I[,4] = as.numeric(M33I[,4])/6 #de pp[mm] a intensidad [mm/h]

```

```

M33I[,5] = as.numeric(M33I[,5])/12 #de pp[mm] a intensidad [mm/h]
M33I[,6] = as.numeric(M33I[,6])/24 #de pp[mm] a intensidad [mm/h]

M33I[1,1] = M33I[1,1]*0.5
M33I[1,2] = M33I[1,2]*1
M33I[1,3] = M33I[1,3]*2
M33I[1,4] = M33I[1,4]*6
M33I[1,5] = M33I[1,5]*12
M33I[1,6] = M33I[1,6]*24

M33O_2 = M33O[-c(3),]
M33O_50 = M33O[-c(2),]
M33O_2 = t(M33O_2)
colnames(M33O_2) = c("d","i")
M33O_50 = t(M33O_50)
colnames(M33O_50) = c("d","i")

M33I_2 = M33I[-c(3),]
M33I_50 = M33I[-c(2),]
M33I_2 = t(M33I_2)
colnames(M33I_2) = c("d","i")
M33I_50 = t(M33I_50)
colnames(M33I_50) = c("d","i")

png("idfM33.png",width = 700, height = 400)

plt_M33 = {plot(M33O_2, type="o", col = "red", lwd=1,font = 2,main = "M0033",
               ylab="", lty=2, xlab = "", cex.lab=2 , cex.main=2 , yaxt = "n",
               xaxt = "n",xlim = c(0, 25), ylim = c(0, 70))
lines(M33I_2, type="o", col = "blue", lty=1 , lwd=1)

```

```

lines(M33O_50, type="o", col = "red3", lty=2, lwd=2 )
lines(M33I_50, type="o", col ="blue4", lty=1 , lwd=2)
title(xlab= "Duración (h)", line=2.3, cex.lab=1.1)
title(ylab= "Intensidad (mm/h)", line=2.3, cex.lab=1.1)
abline(v=Dur, col="snow3", lwd=0.5)
abline(h=yhlabM33, col="snow3", lwd=0.5)
label <- c("Tr 2 years OBS", "Tr 2 years IMERG", "Tr 50 years OBS", "Tr 50 years
IMERG")
legend("topright", label, col = c("red", "blue", "red3", "blue4") ,
      lty=c(2,1,2,1), pch = 1, lwd=1, y.intersp=1, cex = 1 )
axis(1, at = c(0.5,1,2,6,12,24),
      labels = c("0.5", "1", "2", "6", "12", "24"))
axis(2, at = c(20,40,60 ) ,
      labels = c("20", "40", "60"  ))}
dev.off()

#-----
#P12-P27

P12O = read.csv(file="P12EPMAPTOCITUCO.csv")
P27O = read.csv(file="P27EPMAPSANFRANCISCO.csv")
P12I = read.csv(file="P12IMERG.csv")
P12O = P12O[-c(1:25,28,29,31),]
P27O = P27O[-c(1:25,28,29,31),]
P12I = P12I[-c(1:25,28,29,31),]
P12O = P12O[,-c(1)]
P27O = P27O[,-c(1)]
P12I = P12I[,-c(1)]

```

```

colnames(P12O) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")
colnames(P27O) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")
colnames(P12I) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")
P12O[1,] = c("0.5", "1", "2", "6", "12", "24")
P27O[1,] = c("0.5", "1", "2", "6", "12", "24")
P12I[1,] = c("0.5", "1", "2", "6", "12", "24")
P12O[,1] = as.numeric(P12O[,1])/0.5 #de pp[mm] a intensidad [mm/h]
P12O[,2] = as.numeric(P12O[,2])/1 #de pp[mm] a intensidad [mm/h]
P12O[,3] = as.numeric(P12O[,3])/2 #de pp[mm] a intensidad [mm/h]
P12O[,4] = as.numeric(P12O[,4])/6 #de pp[mm] a intensidad [mm/h]
P12O[,5] = as.numeric(P12O[,5])/12 #de pp[mm] a intensidad [mm/h]
P12O[,6] = as.numeric(P12O[,6])/24 #de pp[mm] a intensidad [mm/h]
P27O[,1] = as.numeric(P27O[,1])/0.5 #de pp[mm] a intensidad [mm/h]
P27O[,2] = as.numeric(P27O[,2])/1 #de pp[mm] a intensidad [mm/h]
P27O[,3] = as.numeric(P27O[,3])/2 #de pp[mm] a intensidad [mm/h]
P27O[,4] = as.numeric(P27O[,4])/6 #de pp[mm] a intensidad [mm/h]
P27O[,5] = as.numeric(P27O[,5])/12 #de pp[mm] a intensidad [mm/h]
P27O[,6] = as.numeric(P27O[,6])/24 #de pp[mm] a intensidad [mm/h]
P12I[,1] = as.numeric(P12I[,1])/0.5 #de pp[mm] a intensidad [mm/h]
P12I[,2] = as.numeric(P12I[,2])/1 #de pp[mm] a intensidad [mm/h]
P12I[,3] = as.numeric(P12I[,3])/2 #de pp[mm] a intensidad [mm/h]
P12I[,4] = as.numeric(P12I[,4])/6 #de pp[mm] a intensidad [mm/h]
P12I[,5] = as.numeric(P12I[,5])/12 #de pp[mm] a intensidad [mm/h]
P12I[,6] = as.numeric(P12I[,6])/24 #de pp[mm] a intensidad [mm/h]

```

```

P12O[1,1] = P12O[1,1]*0.5
P12O[1,2] = P12O[1,2]*1
P12O[1,3] = P12O[1,3]*2
P12O[1,4] = P12O[1,4]*6
P12O[1,5] = P12O[1,5]*12
P12O[1,6] = P12O[1,6]*24
P27O[1,1] = P27O[1,1]*0.5
P27O[1,2] = P27O[1,2]*1
P27O[1,3] = P27O[1,3]*2
P27O[1,4] = P27O[1,4]*6
P27O[1,5] = P27O[1,5]*12
P27O[1,6] = P27O[1,6]*24
P12I[1,1] = P12I[1,1]*0.5
P12I[1,2] = P12I[1,2]*1
P12I[1,3] = P12I[1,3]*2
P12I[1,4] = P12I[1,4]*6
P12I[1,5] = P12I[1,5]*12
P12I[1,6] = P12I[1,6]*24
P12O_2 = P12O[-c(3),]
P12O_50 = P12O[-c(2),]
P12O_2 = t(P12O_2)
colnames(P12O_2) = c("d","i")
P12O_50 = t(P12O_50)
colnames(P12O_50) = c("d","i")
P27O_2 = P27O[-c(3),]

```



```

P27O_50 = P27O[-c(2),]
P27O_2 = t(P27O_2)
colnames(P27O_2) = c("d","i")
P27O_50 = t(P27O_50)
colnames(P27O_50) = c("d","i")
P12I_2 = P12I[-c(3),]
P12I_50 = P12I[-c(2),]
P12I_2 = t(P12I_2)
colnames(P12I_2) = c("d","i")
P12I_50 = t(P12I_50)
colnames(P12I_50) = c("d","i")
png("idfP1227.png",width = 700, height = 400)
plt_P3pix = {plot(P12O_2, type="o", col = "red", lwd=1,main = "P12-P27",
  ylab="", lty=2, xlab = "", cex.lab=2 , cex.main=2 ,yaxt = "n",
  xaxt = "n",xlim = c(0, 25), ylim = c(0, 60))
lines(P27O_2, type="o", col = "yellow", lty=2, lwd=1 )
lines(P12I_2, type="o", col = "blue", lty=1 , lwd=1)
lines(P12O_50, type="o", col = "red3", lty=2, lwd=2 )
lines(P27O_50, type="o", col = "yellow2", lty=2, lwd=2 )
lines(P12I_50, type="o", col = "blue4", lty=1 , lwd=2)
title(xlab= "Duración (h)", line=2.3, cex.lab=1.1)
title(ylab= "Intensidad (mm/h)", line=2.3, cex.lab=1.1)
abline(v=Dur, col="snow3", lwd=0.5)
abline(h=yhlabM25, col="snow3", lwd=0.5)
label <- c("Tr 2 years OBS-P12","Tr 2 years OBS-P27","Tr 2 years IMERG","Tr 50 years
OBS-P12","Tr 50 years OBS-P27","Tr 50 years IMERG")

```

```

legend("topright", label, col = c("red", "yellow", "blue", "red3", "yellow2", "blue4"),
      lty=c(2,2,1,2,2,1), pch = 1, lwd=1, y.intersp=1, cex = 1 )
axis(1, at = c(0.5,1,2,6,12,24),
      labels = c("0.5", "1", "2", "6", "12", "24"))
axis(2, at = c(10,20,30,40,50,60),
      labels = c("10", "20", "30", "40", "50", "60"))
}
dev.off()

#-----
#C05-P09-P70

C05O = read.csv(file="C05EPMAPSBELLAVISTA.csv")
P09O = read.csv(file="P09EPMAPSINAQUITOINAMHI.csv")
P70O = read.csv(file="P70EPMAPCCELBOSQUE.csv")
P70I = read.csv(file="P70IMERG.csv")

C05O = C05O[-c(1:25,28,29,31),]
C05O = C05O[,-c(1)]

colnames(C05O) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")
C05O[1,] = c("0.5", "1", "2", "6", "12", "24")

C05O[,1] = as.numeric(C05O[,1])/0.5 #de pp[mm] a intensidad [mm/h]
C05O[,2] = as.numeric(C05O[,2])/1 #de pp[mm] a intensidad [mm/h]
C05O[,3] = as.numeric(C05O[,3])/2 #de pp[mm] a intensidad [mm/h]
C05O[,4] = as.numeric(C05O[,4])/6 #de pp[mm] a intensidad [mm/h]
C05O[,5] = as.numeric(C05O[,5])/12 #de pp[mm] a intensidad [mm/h]
C05O[,6] = as.numeric(C05O[,6])/24 #de pp[mm] a intensidad [mm/h]

C05O[1,1] = C05O[1,1]*0.5

```

```

C05O[1,2] = C05O[1,2]*1
C05O[1,3] = C05O[1,3]*2
C05O[1,4] = C05O[1,4]*6
C05O[1,5] = C05O[1,5]*12
C05O[1,6] = C05O[1,6]*24
C05O_2 = C05O[-c(3),]
C05O_50 = C05O[-c(2),]
C05O_2 = t(C05O_2)
colnames(C05O_2) = c("d","i")
C05O_50 = t(C05O_50)
colnames(C05O_50) = c("d","i")
P09O = P09O[-c(1:25,28,29,31),]
P09O = P09O[-c(1)]
colnames(P09O) = c("0.5hr","1hr","2hr","6hr","12hr","24hr")
P09O[1,] = c("0.5","1","2","6","12","24")
P09O[,1] = as.numeric(P09O[,1])/0.5 #de pp[mm] a intensidad [mm/h]
P09O[,2] = as.numeric(P09O[,2])/1 #de pp[mm] a intensidad [mm/h]
P09O[,3] = as.numeric(P09O[,3])/2 #de pp[mm] a intensidad [mm/h]
P09O[,4] = as.numeric(P09O[,4])/6 #de pp[mm] a intensidad [mm/h]
P09O[,5] = as.numeric(P09O[,5])/12 #de pp[mm] a intensidad [mm/h]
P09O[,6] = as.numeric(P09O[,6])/24 #de pp[mm] a intensidad [mm/h]
P09O[1,1] = P09O[1,1]*0.5
P09O[1,2] = P09O[1,2]*1
P09O[1,3] = P09O[1,3]*2
P09O[1,4] = P09O[1,4]*6

```

```

P09O[1,5] = P09O[1,5]*12
P09O[1,6] = P09O[1,6]*24
P09O_2 = P09O[-c(3),]
P09O_50 = P09O[-c(2),]
P09O_2 = t(P09O_2)
colnames(P09O_2) = c("d","i")
P09O_50 = t(P09O_50)
colnames(P09O_50) = c("d","i")
P70O = P70O[-c(1:25,28,29,31),]
P70O = P70O[-c(1)]
colnames(P70O) = c("0.5hr","1hr","2hr","6hr","12hr","24hr")
P70O[1,] = c("0.5","1","2","6","12","24")
P70O[1,1] = as.numeric(P70O[1,1])/0.5 #de pp[mm] a intensidad [mm/h]
P70O[1,2] = as.numeric(P70O[1,2])/1 #de pp[mm] a intensidad [mm/h]
P70O[1,3] = as.numeric(P70O[1,3])/2 #de pp[mm] a intensidad [mm/h]
P70O[1,4] = as.numeric(P70O[1,4])/6 #de pp[mm] a intensidad [mm/h]
P70O[1,5] = as.numeric(P70O[1,5])/12 #de pp[mm] a intensidad [mm/h]
P70O[1,6] = as.numeric(P70O[1,6])/24 #de pp[mm] a intensidad [mm/h]
P70O[1,1] = P70O[1,1]*0.5
P70O[1,2] = P70O[1,2]*1
P70O[1,3] = P70O[1,3]*2
P70O[1,4] = P70O[1,4]*6
P70O[1,5] = P70O[1,5]*12
P70O[1,6] = P70O[1,6]*24
P70O_2 = P70O[-c(3),]

```

```

P70O_50 = P70O[-c(2),]
P70O_2 = t(P70O_2)
colnames(P70O_2) = c("d","i")
P70O_50 = t(P70O_50)
colnames(P70O_50) = c("d","i")
P70I = P70I[-c(1:25,28,29,31),]
P70I = P70I[-c(1)]
colnames(P70I) = c("0.5hr","1hr","2hr","6hr","12hr","24hr")
P70I[1,] = c("0.5","1","2","6","12","24")
P70I[,1] = as.numeric(P70I[,1])/0.5 #de pp[mm] a intensidad [mm/h]
P70I[,2] = as.numeric(P70I[,2])/1 #de pp[mm] a intensidad [mm/h]
P70I[,3] = as.numeric(P70I[,3])/2 #de pp[mm] a intensidad [mm/h]
P70I[,4] = as.numeric(P70I[,4])/6 #de pp[mm] a intensidad [mm/h]
P70I[,5] = as.numeric(P70I[,5])/12 #de pp[mm] a intensidad [mm/h]
P70I[,6] = as.numeric(P70I[,6])/24 #de pp[mm] a intensidad [mm/h]
P70I[1,1] = P70I[1,1]*0.5
P70I[1,2] = P70I[1,2]*1
P70I[1,3] = P70I[1,3]*2
P70I[1,4] = P70I[1,4]*6
P70I[1,5] = P70I[1,5]*12
P70I[1,6] = P70I[1,6]*24
P70I_2 = P70I[-c(3),]
P70I_50 = P70I[-c(2),]
P70I_2 = t(P70I_2)
colnames(P70I_2) = c("d","i")

```

```

P70I_50 = t(P70I_50)

colnames(P70I_50) = c("d","i")

png("idfC05P09P70.png",width = 700, height = 400)

plt_P4pix = {plot(C05O_2, type="o", col = "red", lwd=1,main = "C05-P09-P70",
  ylab="", lty=2, xlab= "", cex.lab=2 , cex.main=2 ,yaxt = "n",
  xaxt = "n",xlim = c(0, 25), ylim = c(0, 65))

lines(P09O_2, type="o", col = "violetred1", lty=2, lwd=1 )

lines(P70O_2, type="o", col = "yellow", lty=2, lwd=1 )

lines(P70I_2, type="o", col = "blue", lty=1 , lwd=1)

lines(C05O_50, type="o", col = "red3", lty=2, lwd=2 )

lines(P09O_50, type="o", col = "violetred2", lty=2, lwd=2 )

lines(P70O_50, type="o", col = "yellow3", lty=2, lwd=2 )

lines(P70I_50, type="o", col = "blue4", lty=1 , lwd=2)

title(xlab= "Duración (h)", line=2.3, cex.lab=1.1)

title(ylab= "Intensidad (mm/h)", line=2.3, cex.lab=1.1)

abline(v=Dur, col="snow3", lwd=0.5)

abline(h=yhlabM25, col="snow3", lwd=0.5)

label <- c("Tr 2 years OBS-C05","Tr 2 years OBS-P09","Tr 2 years OBS-P70","Tr 2 years
IMERG","Tr 50 years OBS-C05","Tr 50 years OBS-P09","Tr 50 years OBS-P70","Tr 50
years IMERG")

legend("topright", label, col =
c("red","violetred1","yellow","blue","red3","violetred2","yellow3","blue4") ,

  lty=c(2,2,2,1,2,2,2,1), pch = 1, lwd=1, y.intersp=1, cex = 1 )

axis(1, at = c(0.5,1,2,6,12,24),

  labels = c("0.5", "1", "2", "6", "12", "24"))

axis(2, at = c(10,20,30,40,50,60),

```

```

labels = c("10","20","30","40","50","60"))}

dev.off()

#-----

#C02-C04-C14-P03-P08-P11-P28-P31-P32

C02O = read.csv(file="C02_EPMAPSRUMIHURCO.csv")
C04O = read.csv(file="C04EPMAPSRUMIPANBA.csv")
C14O = read.csv(file="C14EPMAPSMINDOCAPTACION .csv")
P03O = read.csv(file="P03EPMAPSRUMIHURCOMACHANGARA.csv")
P08O = read.csv(file="P08EPMAPSRUMIPAMBA.csv")
P11O = read.csv(file="P11EPMAPSANTENAS.csv")
P28O = read.csv(file="P28EPMAPCRUZLOMA.csv")
P31O = read.csv(file="P31EPMAPPICHAN.csv")
P32O = read.csv(file="P32EPMAPMINDOBAJO.csv")
P32I = read.csv(file="P32IMERG.csv")

C02O = C02O[-c(1:25,28,29,31),]
C04O = C04O[-c(1:25,28,29,31),]
C14O = C14O[-c(1:25,28,29,31),]
P03O = P03O[-c(1:25,28,29,31),]
P08O = P08O[-c(1:25,28,29,31),]
P11O = P11O[-c(1:25,28,29,31),]
P28O = P28O[-c(1:25,28,29,31),]
P31O = P31O[-c(1:25,28,29,31),]
P32O = P32O[-c(1:25,28,29,31),]
P32I = P32I[-c(1:25,28,29,31),]

```

C02O = C02O[-c(1)]

C04O = C04O[-c(1)]

C14O = C14O[-c(1)]

P03O = P03O[-c(1)]

P08O = P08O[-c(1)]

P11O = P11O[-c(1)]

P28O = P28O[-c(1)]

P31O = P31O[-c(1)]

P32O = P32O[-c(1)]

P32I = P32I[-c(1)]

colnames(C02O) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")

colnames(C04O) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")

colnames(C14O) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")

colnames(P03O) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")

colnames(P08O) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")

colnames(P11O) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")

colnames(P28O) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")

colnames(P31O) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")

colnames(P32O) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")

colnames(P32I) = c("0.5hr", "1hr", "2hr", "6hr", "12hr", "24hr")

C02O[1,] = c("0.5", "1", "2", "6", "12", "24")

C04O[1,] = c("0.5", "1", "2", "6", "12", "24")

C14O[1,] = c("0.5", "1", "2", "6", "12", "24")

P03O[1,] = c("0.5", "1", "2", "6", "12", "24")

P08O[1,] = c("0.5", "1", "2", "6", "12", "24")

P11O[1,] = c("0.5", "1", "2", "6", "12", "24")
 P28O[1,] = c("0.5", "1", "2", "6", "12", "24")
 P31O[1,] = c("0.5", "1", "2", "6", "12", "24")
 P32O[1,] = c("0.5", "1", "2", "6", "12", "24")
 P32I[1,] = c("0.5", "1", "2", "6", "12", "24")
 C02O[1,] = as.numeric(C02O[,1])/0.5 #de pp[mm] a intensidad [mm/h]
 C02O[2,] = as.numeric(C02O[,2])/1 #de pp[mm] a intensidad [mm/h]
 C02O[3,] = as.numeric(C02O[,3])/2 #de pp[mm] a intensidad [mm/h]
 C02O[4,] = as.numeric(C02O[,4])/6 #de pp[mm] a intensidad [mm/h]
 C02O[5,] = as.numeric(C02O[,5])/12 #de pp[mm] a intensidad [mm/h]
 C02O[6,] = as.numeric(C02O[,6])/24 #de pp[mm] a intensidad [mm/h]
 C04O[1,] = as.numeric(C04O[,1])/0.5 #de pp[mm] a intensidad [mm/h]
 C04O[2,] = as.numeric(C04O[,2])/1 #de pp[mm] a intensidad [mm/h]
 C04O[3,] = as.numeric(C04O[,3])/2 #de pp[mm] a intensidad [mm/h]
 C04O[4,] = as.numeric(C04O[,4])/6 #de pp[mm] a intensidad [mm/h]
 C04O[5,] = as.numeric(C04O[,5])/12 #de pp[mm] a intensidad [mm/h]
 C04O[6,] = as.numeric(C04O[,6])/24 #de pp[mm] a intensidad [mm/h]
 C14O[1,] = as.numeric(C14O[,1])/0.5 #de pp[mm] a intensidad [mm/h]
 C14O[2,] = as.numeric(C14O[,2])/1 #de pp[mm] a intensidad [mm/h]
 C14O[3,] = as.numeric(C14O[,3])/2 #de pp[mm] a intensidad [mm/h]
 C14O[4,] = as.numeric(C14O[,4])/6 #de pp[mm] a intensidad [mm/h]
 C14O[5,] = as.numeric(C14O[,5])/12 #de pp[mm] a intensidad [mm/h]
 C14O[6,] = as.numeric(C14O[,6])/24 #de pp[mm] a intensidad [mm/h]
 P03O[1,] = as.numeric(P03O[,1])/0.5 #de pp[mm] a intensidad [mm/h]
 P03O[2,] = as.numeric(P03O[,2])/1 #de pp[mm] a intensidad [mm/h]

P03O[,3] = as.numeric(P03O[,3])/2 #de pp[mm] a intensidad [mm/h]
 P03O[,4] = as.numeric(P03O[,4])/6 #de pp[mm] a intensidad [mm/h]
 P03O[,5] = as.numeric(P03O[,5])/12 #de pp[mm] a intensidad [mm/h]
 P03O[,6] = as.numeric(P03O[,6])/24 #de pp[mm] a intensidad [mm/h]
 P08O[,1] = as.numeric(P08O[,1])/0.5 #de pp[mm] a intensidad [mm/h]
 P08O[,2] = as.numeric(P08O[,2])/1 #de pp[mm] a intensidad [mm/h]
 P08O[,3] = as.numeric(P08O[,3])/2 #de pp[mm] a intensidad [mm/h]
 P08O[,4] = as.numeric(P08O[,4])/6 #de pp[mm] a intensidad [mm/h]
 P08O[,5] = as.numeric(P08O[,5])/12 #de pp[mm] a intensidad [mm/h]
 P08O[,6] = as.numeric(P08O[,6])/24 #de pp[mm] a intensidad [mm/h]
 P11O[,1] = as.numeric(P11O[,1])/0.5 #de pp[mm] a intensidad [mm/h]
 P11O[,2] = as.numeric(P11O[,2])/1 #de pp[mm] a intensidad [mm/h]
 P11O[,3] = as.numeric(P11O[,3])/2 #de pp[mm] a intensidad [mm/h]
 P11O[,4] = as.numeric(P11O[,4])/6 #de pp[mm] a intensidad [mm/h]
 P11O[,5] = as.numeric(P11O[,5])/12 #de pp[mm] a intensidad [mm/h]
 P11O[,6] = as.numeric(P11O[,6])/24 #de pp[mm] a intensidad [mm/h]
 P28O[,1] = as.numeric(P28O[,1])/0.5 #de pp[mm] a intensidad [mm/h]
 P28O[,2] = as.numeric(P28O[,2])/1 #de pp[mm] a intensidad [mm/h]
 P28O[,3] = as.numeric(P28O[,3])/2 #de pp[mm] a intensidad [mm/h]
 P28O[,4] = as.numeric(P28O[,4])/6 #de pp[mm] a intensidad [mm/h]
 P28O[,5] = as.numeric(P28O[,5])/12 #de pp[mm] a intensidad [mm/h]
 P28O[,6] = as.numeric(P28O[,6])/24 #de pp[mm] a intensidad [mm/h]
 P31O[,1] = as.numeric(P31O[,1])/0.5 #de pp[mm] a intensidad [mm/h]
 P31O[,2] = as.numeric(P31O[,2])/1 #de pp[mm] a intensidad [mm/h]
 P31O[,3] = as.numeric(P31O[,3])/2 #de pp[mm] a intensidad [mm/h]

```

P31O[,4] = as.numeric(P31O[,4])/6 #de pp[mm] a intensidad [mm/h]
P31O[,5] = as.numeric(P31O[,5])/12 #de pp[mm] a intensidad [mm/h]
P31O[,6] = as.numeric(P31O[,6])/24 #de pp[mm] a intensidad [mm/h]
P32O[,1] = as.numeric(P32O[,1])/0.5 #de pp[mm] a intensidad [mm/h]
P32O[,2] = as.numeric(P32O[,2])/1 #de pp[mm] a intensidad [mm/h]
P32O[,3] = as.numeric(P32O[,3])/2 #de pp[mm] a intensidad [mm/h]
P32O[,4] = as.numeric(P32O[,4])/6 #de pp[mm] a intensidad [mm/h]
P32O[,5] = as.numeric(P32O[,5])/12 #de pp[mm] a intensidad [mm/h]
P32O[,6] = as.numeric(P32O[,6])/24 #de pp[mm] a intensidad [mm/h]
P32I[,1] = as.numeric(P32I[,1])/0.5 #de pp[mm] a intensidad [mm/h]
P32I[,2] = as.numeric(P32I[,2])/1 #de pp[mm] a intensidad [mm/h]
P32I[,3] = as.numeric(P32I[,3])/2 #de pp[mm] a intensidad [mm/h]
P32I[,4] = as.numeric(P32I[,4])/6 #de pp[mm] a intensidad [mm/h]
P32I[,5] = as.numeric(P32I[,5])/12 #de pp[mm] a intensidad [mm/h]
P32I[,6] = as.numeric(P32I[,6])/24 #de pp[mm] a intensidad [mm/h]
C02O[1,1] = C02O[1,1]*0.5
C02O[1,2] = C02O[1,2]*1
C02O[1,3] = C02O[1,3]*2
C02O[1,4] = C02O[1,4]*6
C02O[1,5] = C02O[1,5]*12
C02O[1,6] = C02O[1,6]*24
C02O_2 = C02O[-c(3),]
C02O_50 = C02O[-c(2),]
C02O_2 = t(C02O_2)
colnames(C02O_2) = c("d","i")

```

```

C02O_50 = t(C02O_50)
colnames(C02O_50) = c("d","i")

C04O[1,1] = C04O[1,1]*0.5
C04O[1,2] = C04O[1,2]*1
C04O[1,3] = C04O[1,3]*2
C04O[1,4] = C04O[1,4]*6
C04O[1,5] = C04O[1,5]*12
C04O[1,6] = C04O[1,6]*24

C04O_2 = C04O[-c(3),]
C04O_50 = C04O[-c(2),]

C04O_2 = t(C04O_2)
colnames(C04O_2) = c("d","i")

C04O_50 = t(C04O_50)
colnames(C04O_50) = c("d","i")

C14O[1,1] = C14O[1,1]*0.5
C14O[1,2] = C14O[1,2]*1
C14O[1,3] = C14O[1,3]*2
C14O[1,4] = C14O[1,4]*6
C14O[1,5] = C14O[1,5]*12
C14O[1,6] = C14O[1,6]*24

C14O_2 = C14O[-c(3),]
C14O_50 = C14O[-c(2),]

C14O_2 = t(C14O_2)
colnames(C14O_2) = c("d","i")

C14O_50 = t(C14O_50)

```

```

colnames(C14O_50) = c("d","i")
P03O[1,1] = P03O[1,1]*0.5
P03O[1,2] = P03O[1,2]*1
P03O[1,3] = P03O[1,3]*2
P03O[1,4] = P03O[1,4]*6
P03O[1,5] = P03O[1,5]*12
P03O[1,6] = P03O[1,6]*24
P03O_2 = P03O[-c(3),]
P03O_50 = P03O[-c(2),]
P03O_2 = t(P03O_2)
colnames(P03O_2) = c("d","i")
P03O_50 = t(P03O_50)
colnames(P03O_50) = c("d","i")
P08O[1,1] = P08O[1,1]*0.5
P08O[1,2] = P08O[1,2]*1
P08O[1,3] = P08O[1,3]*2
P08O[1,4] = P08O[1,4]*6
P08O[1,5] = P08O[1,5]*12
P08O[1,6] = P08O[1,6]*24
P08O_2 = P08O[-c(3),]
P08O_50 = P08O[-c(2),]
P08O_2 = t(P08O_2)
colnames(P08O_2) = c("d","i")
P08O_50 = t(P08O_50)
colnames(P08O_50) = c("d","i")

```

```

P11O[1,1] = P11O[1,1]*0.5
P11O[1,2] = P11O[1,2]*1
P11O[1,3] = P11O[1,3]*2
P11O[1,4] = P11O[1,4]*6
P11O[1,5] = P11O[1,5]*12
P11O[1,6] = P11O[1,6]*24
P11O_2 = P11O[-c(3),]
P11O_50 = P11O[-c(2),]
P11O_2 = t(P11O_2)
colnames(P11O_2) = c("d","i")
P11O_50 = t(P11O_50)
colnames(P11O_50) = c("d","i")
P28O[1,1] = P28O[1,1]*0.5
P28O[1,2] = P28O[1,2]*1
P28O[1,3] = P28O[1,3]*2
P28O[1,4] = P28O[1,4]*6
P28O[1,5] = P28O[1,5]*12
P28O[1,6] = P28O[1,6]*24
P28O_2 = P28O[-c(3),]
P28O_50 = P28O[-c(2),]
P28O_2 = t(P28O_2)
colnames(P28O_2) = c("d","i")
P28O_50 = t(P28O_50)
colnames(P28O_50) = c("d","i")

```

```

P31O[1,1] = P31O[1,1]*0.5
P31O[1,2] = P31O[1,2]*1
P31O[1,3] = P31O[1,3]*2
P31O[1,4] = P31O[1,4]*6
P31O[1,5] = P31O[1,5]*12
P31O[1,6] = P31O[1,6]*24
P31O_2 = P31O[-c(3),]
P31O_50 = P31O[-c(2),]
P31O_2 = t(P31O_2)
colnames(P31O_2) = c("d","i")
P31O_50 = t(P31O_50)
colnames(P31O_50) = c("d","i")
P32O[1,1] = P32O[1,1]*0.5
P32O[1,2] = P32O[1,2]*1
P32O[1,3] = P32O[1,3]*2
P32O[1,4] = P32O[1,4]*6
P32O[1,5] = P32O[1,5]*12
P32O[1,6] = P32O[1,6]*24
P32O_2 = P32O[-c(3),]
P32O_50 = P32O[-c(2),]
P32O_2 = t(P32O_2)
colnames(P32O_2) = c("d","i")
P32O_50 = t(P32O_50)
colnames(P32O_50) = c("d","i")
P32I[1,1] = P32I[1,1]*0.5

```

```

P32I[1,2] = P32I[1,2]*1
P32I[1,3] = P32I[1,3]*2
P32I[1,4] = P32I[1,4]*6
P32I[1,5] = P32I[1,5]*12
P32I[1,6] = P32I[1,6]*24
P32I_2 = P32I[-c(3),]
P32I_50 = P32I[-c(2),]
P32I_2 = t(P32I_2)
colnames(P32I_2) = c("d","i")
P32I_50 = t(P32I_50)
colnames(P32I_50) = c("d","i")

#-----
# COMPLETO
#-----

png("idfC02 999999.png",width = 700, height = 400)

plt_P9pix = {plot(C02O_2, type="o", col = "red", lwd=1,main = "C02-C04-C14-P03-P08-
P11-P28-P31-P32",
  ylab="", lty=2, xlab= "", cex.lab=2 , cex.main=2 ,yaxt = "n",
  xaxt = "n",xlim = c(0, 25), ylim = c(0, 60))

lines(C04O_2, type="o", col = "#6E8B3D", lty=2, lwd=1 )
lines(C14O_2, type="o", col = "darkgoldenrod1", lty=2, lwd=1 )
lines(P03O_2, type="o", col = "#00FF00", lty=2, lwd=1 )
lines(P08O_2, type="o", col = "#FFF68F", lty=2, lwd=1 )
lines(P11O_2, type="o", col = "#EE9572", lty=2, lwd=1 )
lines(P28O_2, type="o", col = "violetred1", lty=2, lwd=1 )
lines(P31O_2, type="o", col = "yellow", lty=2, lwd=1 )

```



```

lines(P32O_2, type="o", col = "#B23AEE", lty=2, lwd=1 )
lines(P32I_2, type="o", col = "blue", lty=1 , lwd=1)
lines(C02O_50, type="o", col = "red3", lty=2, lwd=2 )
lines(C04O_50, type="o", col = "#6E8B3D", lty=2, lwd=2 )
lines(C14O_50, type="o", col = "darkgoldenrod1", lty=2, lwd=2 )
lines(P03O_50, type="o", col = "#00FF00", lty=2, lwd=2 )
lines(P08O_50, type="o", col = "#FFF68F", lty=2, lwd=2 )
lines(P11O_50, type="o", col = "#EE9572", lty=2, lwd=2 )
lines(P28O_50, type="o", col = "violetred", lty=2, lwd=2 )
lines(P31O_50, type="o", col = "yellow", lty=2, lwd=2)
lines(P32O_50, type="o", col = "#B23AEE", lty=2, lwd=2 )
lines(P32I_50, type="o", col = "#FF3E96", lty=1 , lwd=2)
lines(P28O_50, type="o", col = "#5CACEE", lty=2, lwd=2 )
lines(P31O_50, type="o", col = "#6E8B3D", lty=2, lwd=2 )
lines(P32O_50, type="o", col = "#B23AEE", lty=2, lwd=2 )
lines(P32I_50, type="o", col = "blue4", lty=1 , lwd=2)
title(xlab= "Duración (h)", line=2.3, cex.lab=1.1)
title(ylab= "Intensidad (mm/h)", line=2.3, cex.lab=1.1)
abline(v=Dur, col="snow3", lwd=0.5)
abline(h=yhlabM25, col="snow3", lwd=0.5)
label <- c("Tr 2 years OBS-C02", "Tr 2 years OBS-C04", "Tr 2 years OBS-C14", "Tr 2 years
OBS-P03", "Tr 2 years OBS-P08", "Tr 2 years OBS-P11", "Tr 2 years OBS-P28", "Tr 2 years
OBS-P31", "Tr 2 years OBS-P32", "Tr 2 years IMERG", "Tr 50 years OBS-C02", "Tr 50 years
OBS-C04", "Tr 50 years OBS-C14", "Tr 50 years OBS-P03", "Tr 50 years OBS-P08", "Tr 50
years OBS-P11", "Tr 50 years OBS-P28", "Tr 50 years OBS-P31", "Tr 50 years OBS-
P32", "Tr 50 years IMERG")

```

```

legend("topright", label, col =
c("red", "#6E8B3D", "darkgoldenrod1", "#00FF00", "#FFF68F", "#EE9572", "violetred1", "yellow", "#B23AEE", "blue", "red3", "#6E8B3D", "darkgoldenrod1", "#00FF00", "#FFF68F", "#EE9572", "violetred", "yellow", "#B23AEE", "blue4") ,

      lty=c(2,2,2,2,2,2,2,2,1,2,2,2,2,2,2,2,1), pch = 1, lwd=1, cex = 0.7 )

axis(1, at = c(0.5, 1, 2, 6, 12, 24),

      labels = c("0.5", "1", "2", "6", "12", "24"))

axis(2, at = c(10, 20, 30, 40, 50, 60),

      labels = c("10", "20", "30", "40", "50", "60"))}

dev.off()

#-----

png(file='fig9_.png', width = 1000, height = 600)

plt_fig3

dev.off()

#####

```