Universidad del Bío-Bío

Facultad de Ciencias

Ciencia de Datos en la Terminal Linux
Certamen 1: Clasificación Sísmica

Esteban Saavedra

Profesor Luis Gómez

2 de noviembre de 2025

Índice

Índice de figuras

Índice de cuadros

1. Introducción

La presente evaluación aborda el análisis y modelamiento de datos sísmicos utilizando exclusivamente herramientas de la terminal Linux. El objetivo central, planteado por la red Sismo Data, es determinar qué **señales observables en los registros sísmicos se asocian a eventos de mayor magnitud o riesgo**.

Para lograrlo, se emplea un flujo de trabajo de Ciencia de Datos que abarca la obtención, limpieza, exploración, visualización y modelamiento de los registros. La variable objetivo para el modelamiento predictivo es el **tipo de evento** (`type`), y se utilizan diversas herramientas como `curl`, la suite `csvkit`, `xsv`, `cols`, y scripts de `Python` ejecutados vía `skll` y `rush plot` para procesar y analizar el dataset.

Este enfoque metodológico garantiza la **reproducibilidad** y la **eficiencia** en el tratamiento de grandes volúmenes de datos, cumpliendo con los estándares de la administración de sistemas en entornos de Ciencia de Datos.

2. Metodología

La metodología se estructura en las fases de obtención, limpieza y transformación del conjunto de datos sísmicos.

2.1. Obtención y Muestreo de Datos

El conjunto de datos inicial fue descargado desde una fuente externa (USGS) y se muestrearon 1000 registros para facilitar el proceso de depuración y modelamiento.

Lo primero que debemos hacer es descargar los datos en el servidor linux. Para ello aplicamos:

```
# Descargamos los datos
curl -L \
"https://www.dropbox.com/scl/fi/acm84xjyrj5xlz77ffdpp/data.csv?rlkey=lx4hovpvtyxtkaqoww2vkliyz&st=ffx9borz&dl=0" > data.csv
```

Y luego visualizamos la estructura inicial con `csvlook`:

```
csvlook data.csv | trim
```

```
$ csvlook data.csv | trim
|                          time | latitude | longitude |    depth |     mag | magType | nst |      gap |    dmin |    rms | net | id          |…
| ------------------------------ | -------- | --------- | -------- | ------- | ------- | --- | -------- | ------- | ------ | --- | ----------- |…
| 2025-10-17 19:27:56.890000+00:00 |  38.792… | -122.780… |   2.680… |  0.900… | md      |   8 | 154.000… |  0.007… | 0.020… | nc  | …
| 2025-10-17 19:03:22.395000+00:00 |  61.558… | -146.541… |  44.100… |  1.500… | ml      |     |          |         | 1.010… | ak  | ak02…
| 2025-10-17 18:58:12.600000+00:00 |  61.690… | -149.889… |  29.100… |  1.700… | ml      |     |          |         | 0.650… | ak  | ak02…
| 2025-10-17 18:51:19.660000+00:00 |  39.129… | -104.640… |   7.811… |  3.000… | ml      |  15 |  65.000… |  0.429… | 0.940… | us  | …
| 2025-10-17 18:26:20.395000+00:00 |  57.254… | -155.644… |  71.800… |  1.900… | ml      |     |          |         | 0.190… | ak  | ak02…
| 2025-10-17 18:20:46.982000+00:00 |  38.023… | -116.546… |   7.323… |  2.430… | ml      |  15 | 299.000… |  0.830… | 0.322… | nn  | …
| 2025-10-17 18:02:03.710000+00:00 |  35.642… | -117.459… |   5.830… |  0.670… | ml      |  16 |  75.000… |  0.037… | 0.110… | ci  | …
| 2025-10-17 18:00:31.850000+00:00 |  29.045… |  -98.315… |   3.956… |  1.900… | ml      |   4 | 266.000… |  0.200… | 0.200… | tx  | …
… with 9069 more lines
```

Figura 1: Visualización inicial de la estructura del dataset con `csvlook`.

Podemos ver el listado de todas las variables que contiene el conjunto de datos, usando:

```
csvcut -n data.csv
```

Figura 2: Listado de variables y sus índices en el conjunto de datos original.

Ahora realizamos una muestra de 1000 registros:

```
xsv sample 1000 data.csv > sample_earthquakes.csv
xsv count sample_earthquakes.csv
```

2.2. Limpieza y Transformación de Datos

Esta fase se centró en la imputación de valores nulos, la estandarización de variables categóricas y la eliminación de columnas irrelevantes para el modelamiento.

2.2.1. Tratamiento de Datos Faltantes (Imputación)

Se identificaron los datos faltantes con `csvstat` y se aplicó imputación por la mediana para las variables numéricas y por la etiqueta 'desconocido' para la variable `place`, utilizando un script de Python (`imputacion.py`).

```
csvstat sample_earthquakes.csv --nulls
```

Variable	¿Contiene valores faltantes?
time	NO
latitude	NO
longitude	NO
depth	NO
mag	NO
magType	NO
nst	SI
gap	SI
dmin	SI
rms	NO
net	NO
id	NO
updated	NO
place	SI
type	NO
horizontalError	SI
depthError	SI
magError	SI
magNst	SI
status	NO
locationSource	NO
magSource	NO

Cuadro 1: Listado de NA's presentes en cada Variable.

Realizamos una imputación, y para ello utilizaremos código python:

```
nano imputación.py
python imputación.py
csvstat earthquakes_imputed.csv --nulls
```

Código python utilizado para la imputación:

```python
import pandas as pd

df = pd.read_csv("sample_earthquakes.csv")

# Imputar por tipo de variable
num_cols = ['nst', 'gap', 'dmin', 'horizontalError', 'depthError', 'magError', 'magNst']
for col in num_cols:
    df[col].fillna(df[col].median(), inplace=True)

# Para variables categóricas
df['place'].fillna('desconocido', inplace=True)

# Guardar
```

```
df.to_csv("earthquakes_imputed.csv", index=False)
```

```
$ csvstat earthquakes_imputed.csv --nulls
  1. time: False
  2. latitude: False
  3. longitude: False
  4. depth: False
  5. mag: False
  6. magType: False
  7. nst: False
  8. gap: False
  9. dmin: False
 10. rms: False
 11. net: False
 12. id: False
 13. updated: False
 14. place: False
 15. type: False
 16. horizontalError: False
 17. depthError: False
 18. magError: False
 19. magNst: False
 20. status: False
 21. locationSource: False
 22. magSource: False
```

Figura 3: Ausencia de Valores Faltantes después de la imputación por la mediana.

2.2.2. Eliminación de Columnas y Estandarización

Se eliminaron columnas consideradas irrelevantes para la clasificación (ej., `id`, `updated`, `magSource`, etc.). Posteriormente, las variables categóricas (`magType` y `type`) se estandarizaron a minúsculas usando la herramienta `cols` para evitar errores en el modelamiento.

Elimino columnas irrelevantes para la clasificación:

```
csvcut -C 1,14,20,11,12,13,21,22 earthquakes_imputed.csv > earthquakes_temp_dropped.csv
```

Ahora estandarizamos las columnas de tipo texto a minúscula:

```
# Para columna magType
cat earthquakes_temp_dropped.csv | \
cols -c magType body "tr '[:upper:]' '[:lower:]'" \
> temp_lower_1.csv
# Para columna type
cat temp_lower_1.csv | \
cols -c type body "tr '[:upper:]' '[:lower:]'" \
> temp_lower_2.csv
# Archivo Limpio
mv temp_lower_2.csv earthquakes_clean.csv
```

3. Resultados y Análisis

3.1. Estadísticas Descriptivas por Tipo de Magnitud

El análisis descriptivo se centró en la variable `magType` para entender cómo las diferentes metodologías de medición se asocian a las variables clave (magnitud, profundidad, etc.).

```
import pandas as pd

# Cargar datos limpios
df = pd.read_csv("earthquakes_clean.csv")

# Agrupar por magType y calcular estadísticas descriptivas
stats = df.groupby("magType").agg({
    'latitude': ['mean', 'median', 'min', 'max', 'std'],
    'longitude': ['mean', 'median', 'min', 'max', 'std'],
    'depth': ['mean', 'median', 'min', 'max', 'std'],
    'mag': ['mean', 'median', 'min', 'max', 'std'],
    'nst': ['mean', 'median', 'min', 'max', 'std'],
    'gap': ['mean', 'median', 'min', 'max', 'std'],
    'dmin': ['mean', 'median', 'min', 'max', 'std'],
    'rms': ['mean', 'median', 'min', 'max', 'std'],
    'horizontalError': ['mean', 'median', 'min', 'max', 'std'],
    'depthError': ['mean', 'median', 'min', 'max', 'std'],
    'magError': ['mean', 'median', 'min', 'max', 'std'],
    'magNst': ['mean', 'median', 'min', 'max', 'std']
})

# Aplanar las columnas MultiIndex
stats.columns = ['_'.join(col).strip() for col in stats.columns.values]

# Transponer para que cada variable quede hacia abajo
stats_t = stats.transpose()

print(stats_t)
```

Lo aplico en la consola:

```
python estadisticas.py
```

Cuadro 2: Estadísticas Descriptivas de Variables Sísmicas Agrupadas por Tipo de Magnitud (`magType`) Parte 1

Variable	Estadística	Tipos de Magnitudes							
		mb	**mb_lg**	**md**	**ml**	**mlv**	**mw**	**mwr**	**mww**
	Media	22.7034	42.8322	36.6460	43.0894	28.8490	34.3212	10.3444	32.3041
	Mediana	37.7247	42.8322	38.8055	36.7594	28.8820	34.3212	10.3444	39.2551
`latitude`	Mínimo	-60.0310	39.5119	17.8057	19.0220	28.6810	34.3212	10.3444	-21.8800
	Máximo	53.5670	46.1525	49.3060	67.7426	28.9840	34.3212	10.3444	58.7935
	Desv. estándar	33.7944	4.6956	6.7469	12.2525	0.1542	–	–	24.3394
	Media	102.536094	-87.304550	-115.769751	-123.472974	-98.258333	-116.857000	-86.5812	55.365646
	Mediana	152.526600	-87.304550	-122.764000	-117.013250	-98.250000	-116.857000	-86.5812	122.150500
`longitude`	Mínimo	-179.493600	-99.319000	-155.364333	-178.812167	-98.360000	-116.857000	-86.5812	-178.019900
	Máximo	179.855000	-75.290100	-64.623000	178.467667	-98.165000	-116.857000	-86.5812	159.986100
	Desv. estándar	96.190703	16.990998	17.600961	29.776309	0.097767	–	–	118.382499
	Media	56.206978	9.632500	7.027937	13.020050	8.029767	5.220000	19.4260	34.276692
	Mediana	21.154000	9.632500	3.160000	7.077000	6.751900	5.220000	19.4260	22.899000
`depth`	Mínimo	6.850000	9.265000	-0.670000	-3.170000	5.033500	5.220000	19.4260	10.000000
	Máximo	556.025000	10.000000	90.000000	203.400000	12.303900	5.220000	19.4260	128.009000
	Desv. estándar	93.720610	0.519723	11.717873	21.378007	3.799919	–	–	32.862479
	Media	4.608989	2.850000	1.374868	1.195028	1.400000	3.470000	4.2000	5.423077
	Mediana	4.600000	2.850000	1.240000	1.215000	1.400000	3.470000	4.2000	5.300000
`mag`	Mínimo	3.700000	2.800000	-0.260000	-1.650000	1.300000	3.470000	4.2000	4.800000
	Máximo	5.700000	2.900000	4.320000	4.600000	1.500000	3.470000	4.2000	7.400000
	Desv. estándar	0.355681	0.070711	0.770507	0.856115	0.100000	–	–	0.675961
	Media	46.516854	25.500000	18.460317	23.792023	11.000000	112.000000	98.0000	91.692308
	Mediana	40.000000	25.500000	14.000000	20.000000	11.000000	112.000000	98.0000	88.000000
`nst`	Mínimo	15.000000	19.000000	0.000000	4.000000	8.000000	112.000000	98.0000	47.000000
	Máximo	138.000000	32.000000	125.000000	119.000000	14.000000	112.000000	98.0000	147.000000
	Desv. estándar	23.667254	9.192388	15.538722	17.687074	3.000000	–	–	32.198304
	Media	111.067416	71.500000	109.063492	100.471125	76.666667	25.000000	153.0000	74.076923
	Mediana	118.000000	71.500000	91.000000	88.000000	68.000000	25.000000	153.0000	64.000000
`gap`	Mínimo	33.000000	50.000000	17.000000	15.000000	61.000000	25.000000	153.0000	22.000000
	Máximo	182.000000	93.000000	299.000000	343.000000	101.000000	25.000000	153.0000	167.000000
	Desv. estándar	35.923782	30.405592	61.798198	59.103147	21.361960	–	–	40.633446
	Media	2.391921	0.852000	0.064241	0.087124	0.100000	0.036270	1.1830	2.500923
	Mediana	1.441000	0.852000	0.016200	0.059710	0.100000	0.036270	1.1830	1.634000
`dmin`	Mínimo	0.107000	0.517000	0.000757	0.000000	0.100000	0.036270	1.1830	0.559000
	Máximo	20.535000	1.187000	0.791000	1.650000	0.100000	0.036270	1.1830	7.710000
	Desv. estándar	2.789893	0.473762	0.130973	0.125298	0.000000	–	–	2.209079
	Media	0.823371	0.500000	0.078413	0.246016	0.466667	0.150000	0.5300	0.961538
	Mediana	0.780000	0.500000	0.040000	0.172600	0.500000	0.150000	0.5300	1.040000
`rms`	Mínimo	0.430000	0.370000	0.010000	0.010000	0.400000	0.150000	0.5300	0.390000
	Máximo	1.390000	0.630000	0.690000	1.440000	0.500000	0.150000	0.5300	1.480000
	Desv. estándar	0.217746	0.183848	0.098799	0.222156	0.057735	–	–	0.298799

Cuadro 3: Estadísticas Descriptivas de Variables Sísmicas Agrupadas por Tipo de Magnitud (`magType`) Parte 2

Variable	Estadística	Tipos de Magnitudes							
		mb	**mb_lg**	**md**	**ml**	**mlv**	**mw**	**mwr**	**mww**
`horizontalError`	Media	9.135056	4.055000	0.612487	0.609844	0.000000	0.090000	6.4800	7.844615
	Mediana	9.420000	4.055000	0.340000	0.480000	0.000000	0.090000	6.4800	7.690000
	Mínimo	1.830000	3.290000	0.080000	0.000000	0.000000	0.090000	6.4800	3.970000
	Máximo	14.480000	4.820000	5.580000	11.280000	0.000000	0.090000	6.4800	12.660000
	Desv. estándar	2.561498	1.081873	0.825171	0.753416	0.000000	–	–	2.296830
`depthError`	Media	4.477202	4.912500	1.628307	1.843390	4.120382	0.360000	4.3010	3.567077
	Mediana	3.319000	4.912500	0.710000	0.694063	3.702148	0.360000	4.3010	1.922000
	Mínimo	0.972000	1.998000	0.120000	0.100000	3.425988	0.360000	4.3010	1.754000
	Máximo	14.339000	7.827000	31.610000	32.600000	5.233011	0.360000	4.3010	8.191000
	Desv. estándar	2.954309	4.121725	3.605714	4.910220	0.973408	–	–	2.138767
`magError`	Media	0.094663	0.073000	0.183047	0.181671	0.166667	0.159000	0.0830	0.067692
	Mediana	0.092000	0.073000	0.160000	0.159000	0.200000	0.159000	0.0830	0.071000
	Mínimo	0.021000	0.057000	0.000393	0.000000	0.100000	0.159000	0.0830	0.037000
	Máximo	0.356000	0.089000	0.616458	1.170000	0.200000	0.159000	0.0830	0.089000
	Desv. estándar	0.048090	0.022627	0.104694	0.103192	0.057735	–	–	0.018830
`magNst`	Media	81.764045	58.000000	15.820106	15.002849	8.666667	6.000000	14.0000	27.692308
	Mediana	36.000000	58.000000	11.000000	12.500000	9.000000	6.000000	14.0000	19.000000
	Mínimo	2.000000	34.000000	2.000000	1.000000	6.000000	6.000000	14.0000	12.000000
	Máximo	756.000000	82.000000	94.000000	190.000000	11.000000	6.000000	14.0000	69.000000
	Desv. estándar	138.517723	33.941125	13.525367	15.638457	2.516611	–	–	19.934830

El análisis descriptivo agrupado por `magType` (Cuadros 2 y 3) es fundamental, ya que revela el comportamiento de las variables numéricas bajo diferentes métodos de medición sísmica. Se observa una alta variabilidad en las medidas geográficas (`latitude`, `longitude`) y de errores, como lo evidencia la alta Desviación Estándar en la mayoría de los grupos.

En cuanto a la magnitud (`mag`), el tipo `mww` reporta la magnitud máxima (7,4000), mientras que los tipos `md` y `ml` tienen la mayor frecuencia de registro (ver Figura 7), aunque con magnitudes mucho menores (media de 1,3748 y 1,1950, respectivamente). Además, se destaca que los eventos medidos con `mw` y `mwr` muestran una profundidad media relativamente superficial (alrededor de 5 km y 19,4 km).

3.2. Visualización Exploratoria

Se generaron diversas visualizaciones utilizando la herramienta `rush plot` y scripts de Python para explorar distribuciones, relaciones entre variables y el comportamiento geográfico de los sismos.

3.2.1. Distribuciones y Relaciones

```
# Distribución de las magnitudes
rush plot --x mag --geom density --title 'Distribución de Magnitud' earthquakes_clean.csv > plot_mag_density.png
# Distribución de las profundidades
rush plot --x depth --geom density --title 'Distribución de Profundidad' earthquakes_clean.csv > plot_depth_density.png
```

```
rush plot --x depth --y mag --geom smooth --title 'Relación Magnitud vs. Profundidad' earthquakes_clean.csv > plot_mag_vs_depth.png
```

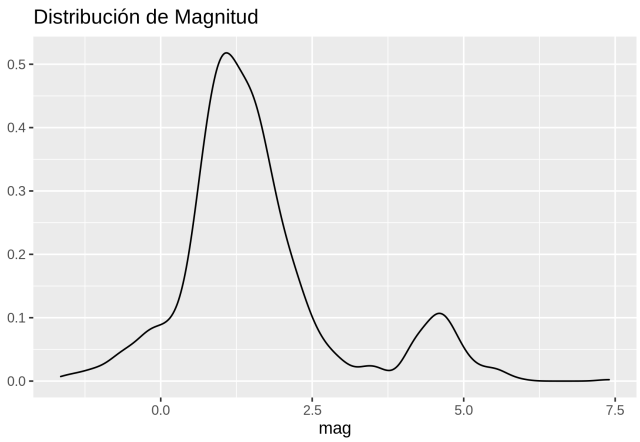

Figura 4: Distribución de la variable `mag`.

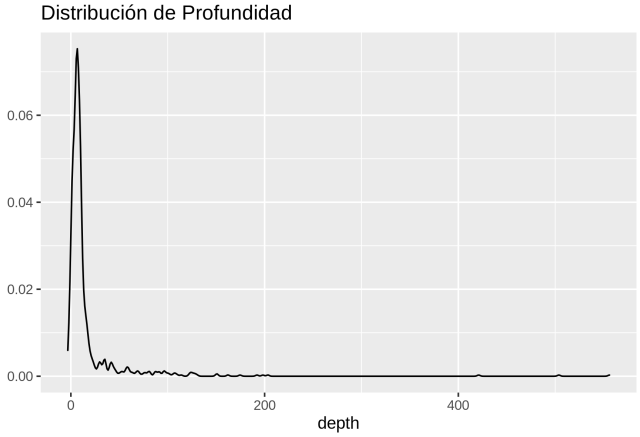

Figura 5: Distribución de la variable `depth`.

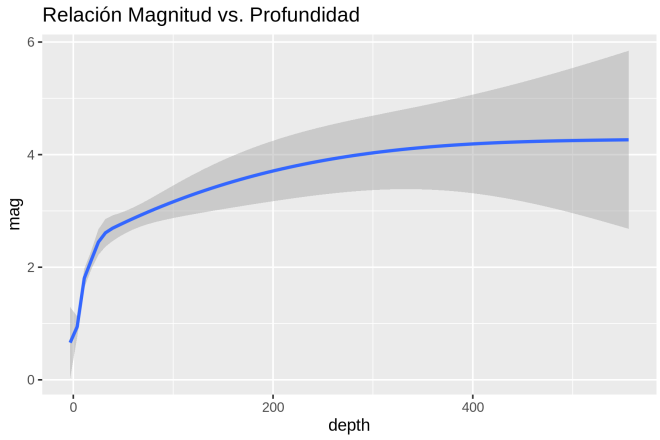

Figura 6: Relación entre las variables `mag` y `depth`.

Las distribuciones univariadas (Figuras 4 y 5) indican que la mayoría de los sismos en la muestra poseen una magnitud baja (alrededor de 1,5 a 2,0) y se concentran en profundidades muy superficiales (cercanas a 0 km).

Esta alta concentración en valores bajos sesga el análisis y subraya la importancia de los eventos con magnitudes o profundidades extremas.

La relación entre Magnitud vs. Profundidad (Figura 6) muestra una correlación positiva. La magnitud tiende a incrementarse logarítmicamente a medida que aumenta la profundidad. Esto sugiere que los sismos más profundos están, en promedio, asociados a magnitudes más altas, aunque con una incertidumbre (sombra gris) que crece con la profundidad.

3.2.2. Comparación por Tipos Categóricos

Debido a fallos con `rush plot`, la visualización de frecuencias por tipo de magnitud (`magType`) se realizó con `matplotlib` en Python.

Gráfico de barras para frecuencia:

```
nano plot_magtype_frequency.py
python plot_magtype_frequency.py
```

Código en python del nano utilizado:

```python
import pandas as pd
import matplotlib.pyplot as plt

# 1. Cargar el archivo limpio
df = pd.read_csv("earthquakes_clean.csv")

# 2. Calcular la frecuencia de cada tipo de magnitud
frequency_data = df['magType'].value_counts().sort_values(ascending=False)

# 3. Generar el gráfico de barras
plt.figure(figsize=(10, 6))

# Crear el gráfico
frequency_data.plot(kind='bar', color='skyblue')

# Configurar etiquetas y título
plt.title('Frecuencia de Tipos de Magnitud (magType)')
plt.xlabel('Tipo de Magnitud (magType)')
```

Boxplot comparativo de Profundidad vs. Tipo de Evento:

```
rush plot --x type --y depth --geom boxplot \
--title 'Profundidad por Tipo de Evento' earthquakes_clean.csv > plot_type_depth_box.png
```

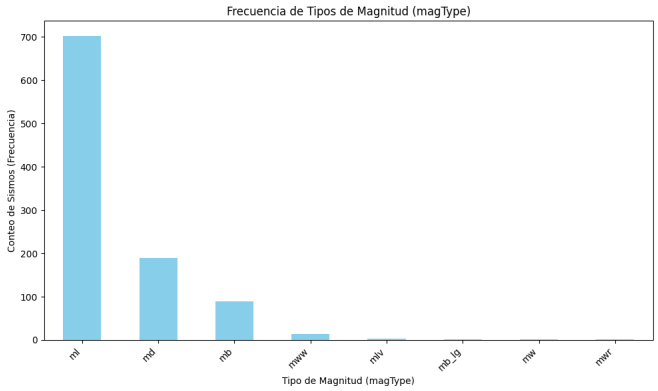

Figura 7: Frecuencias de tipos de magnitudes (`magType`).

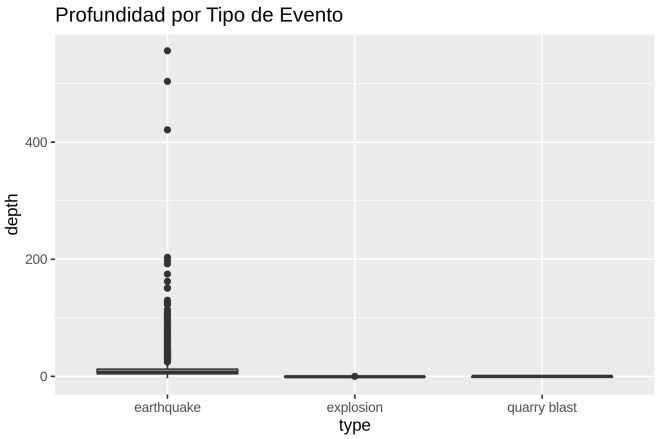

Figura 8: Boxplot `type` vs `depth`.

El gráfico de barras (Figura 7) confirma el predominio absoluto del tipo de medición `ml` (Magnitud Local) en la muestra, seguido por `md` y `mb`. Esto es un factor clave en el modelamiento, ya que el desbalance de clases puede influir en la predicción.

El Boxplot (Figura 8), comparando `type` vs. `depth`, revela una diferencia crítica: si bien los eventos catalogados como `explosion` y `quarry blast` están restringidos casi exclusivamente a profundidades muy superficiales (cercanas a 0 km), los eventos `earthquake` (terremotos) presentan una dispersión extrema en profundidad, alcanzando los 400 km o más, lo que indica una clara separación física en los orígenes de los eventos.

3.2.3. Distribución Geográfica

```
rush plot --x longitude --y latitude --color mag \
--title 'Ubicación de Sismos (Color por Magnitud)' earthquakes_clean.csv > plot_geographic.png
```

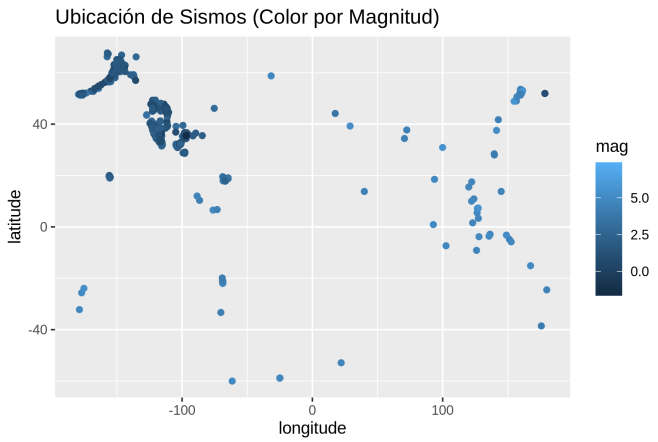

Figura 9: Ubicación de los Sismos y magnitudes de cada uno.

El diagrama de dispersión geográfico (Figura 9) muestra que la mayoría de los sismos se concentran en zonas tectónicamente activas (como la costa oeste de Norteamérica). Es notable que los sismos de mayor magnitud (colores más oscuros) no se distribuyen uniformemente, sino que aparecen como outliers en la nube principal de sismos pequeños, destacando las regiones de alto riesgo.

3.3. Modelamiento Predictivo

Se entrenaron y evaluaron 5 modelos de clasificación (`KNeighborsClassifier`, `LogisticRegression`, `DecisionTreeClassifier`, `RandomForestClassifier`, y `SupportVectorClassifier`) para predecir la variable objetivo `type`, siguiendo un esquema de entrenamiento y prueba (80 %/20 %).

3.3.1. Preparación y Configuración del Experimento SKLL

Los datos se dividieron usando herramientas de la terminal, como `shuf` y `split`, y se configuró el experimento con `classify.cfg`.

```
# Creamos carpetas para entrenamiento y testeo
mkdir -p {train,test}

HEADER="$(< earthquakes_clean.csv header)"
< earthquakes_clean.csv header -d | shuf | split -d -n r/5 - earthquake-part-
wc -l earthquake-part-*
cat earthquake-part-00 | header -a $HEADER > test/features.csv && rm earthquake-part-00
cat earthquake-part-* | header -a $HEADER > train/features.csv && rm earthquake-part-*
wc -l t*/features.csv

nano classify.cfg
skll -l classify.cfg 2>/dev/null
ls -1 output

# Modelos ordenados por accuracy (Más alto primero)
cat output/sismo_clasificacion_summary.tsv |\
csvsql --query "SELECT learner_name, accuracy FROM stdin ORDER BY accuracy DESC" | csvlook -I
# Matrices de confusión para los modelos
jq -r '.[] | "\(.learner_name):\n\(.result_table)\n"' output/*.json
```

Código del nano (`classify.cfg`) utilizado:

```
[General]
experiment_name = sismo_clasificacion
task = evaluate

[Input]
train_directory = train
test_directory = test
featuresets = [["features"]]
feature_scaling = both
label_col = type
shuffle = true
learners = ["KNeighborsClassifier", "LogisticRegression", "DecisionTreeClassifier", "RandomForestClassifier", "SVC"]
suffix = .csv

[Tuning]
grid_search = false
objectives = ["neg_mean_squared_error"]
param_grids = [{}, {}, {}, {}, {}]

[Output]
logs = output
results = output
predictions = output
models = output
```

3.3.2. Análisis de Métricas y Selección del Mejor Modelo

Modelo	Accuracy
RandomForestClassifier	1.000
KNeighborsClassifier	0.995
LogisticRegression	0.995
SupportVectorClassifier	0.995
DecisionTreeClassifier	0.990

Cuadro 4: Métrica Accuracy de los modelos de clasificación.

```
RandomForestClassifier:
+--------------+--------------+-------------+---------------+-------------+----------+-------------+
|              |   earthquake |   explosion |   quarry blast |   Precision |   Recall |   F-measure |
+==============+==============+=============+===============+=============+==========+=============+
|   earthquake |        [197] |           0 |             0 |       1.000 |    1.000 |       1.000 |
+--------------+--------------+-------------+---------------+-------------+----------+-------------+
|    explosion |            0 |         [3] |             0 |       1.000 |    1.000 |       1.000 |
+--------------+--------------+-------------+---------------+-------------+----------+-------------+
| quarry blast |            0 |           0 |           [0] |       0.000 |    0.000 |       0.000 |
+--------------+--------------+-------------+---------------+-------------+----------+-------------+
(row = reference; column = predicted)
```

Figura 10: Matriz de Confusión para el mejor modelo, `RandomForestClassifier`.

La tabla de Métrica Accuracy (Cuadro 4) muestra que el modelo con mejor desempeño fue el `RandomForestClassifier` con una precisión perfecta de 1,000 en el conjunto de prueba. Los modelos `KNeighborsClassifier`, `LogisticRegression`, y `SVC` también lograron una precisión muy alta (0,995), mientras que el `DecisionTreeClassifier` quedó ligeramente rezagado (0,990).

Esto indica que la relación entre las características sísmicas y la clasificación del evento (`type`) es casi linealmente separable en el espacio de características, lo que explica la alta precisión lograda por todos los modelos.

La matriz de confusión del mejor modelo, `RandomForestClassifier` (Figura 10), revela por qué la precisión es tan alta: el modelo clasificó correctamente 197 de los 197 terremotos y los 3 eventos de tipo `explosion`. Sin embargo, la clase `quarry blast` fue clasificada como 0, lo que indica que esta clase probablemente no estaba presente en el conjunto de prueba, o bien fue mal clasificada en su totalidad. El F-measure perfecto (1,000) para `earthquake` y `explosion` confirma la robustez del modelo para esas clases.

4. Conclusiones

El análisis del conjunto de datos sísmicos, ejecutado enteramente en la terminal Linux, permitió establecer hallazgos metodológicos y predictivos cruciales. Primero, en la fase de limpieza, se demostró la robustez del flujo de trabajo al combinar herramientas como `csvkit` y `cols` con scripts de Python/pandas para realizar la imputación por mediana y la estandarización de las variables categóricas, garantizando así un conjunto de datos robusto y completo para el modelamiento.

Respecto al análisis descriptivo, los resultados confirmaron que la Magnitud y la Profundidad son los principales factores sísmicos. La Magnitud máxima fue registrada por el tipo `mww` (7,4000), mientras que la mayor frecuencia de eventos corresponde al tipo `ml`. Además, el análisis de dispersión reveló que la magnitud tiende a incrementarse logarítmicamente con la profundidad, sugiriendo que los sismos más profundos están asociados a mayores liberaciones de energía.

En términos de predicción, se estableció una diferenciación clara en el origen de los eventos: los de bajo riesgo, como las explosiones de cantera, son estrictamente superficiales, mientras que los terremotos (`earthquake`) presentan una dispersión amplia en profundidad. Este hallazgo fue validado por el modelamiento, donde el `RandomForestClassifier` se consolidó como el modelo con el mejor desempeño predictivo para la variable `type`, alcanzando una precisión perfecta (**1,000**) en el conjunto de prueba. En definitiva, las señales clave asociadas a eventos de riesgo son la profundidad variable (no superficial) y las altas desviaciones estándar en las variables de medición, lo que subraya la eficiencia del entorno Linux para ejecutar proyectos de Ciencia de Datos altamente reproducibles.