

Exploración de las relaciones entre aguas subterráneas y variables edáficas

Pedro Bonacic Vera

Table of contents

Contexto	1
Análisis exploratorio	2
Desafíos	11

Contexto

Los humedales altoandinos son piezas clave de los balances de agua, energía y carbono en la región del Altiplano, procesos que dependen de la disponibilidad de aguas subterráneas cercanas a la superficie.

Pese a su relevancia, estos ecosistemas enfrentan amenazas crecientes debido al cambio climático y la actividad minera. Además, los métodos de monitoreo hidrogeológico actuales son caros y difíciles de implementar en estos ambientes remotos.

Por tanto, el objetivo de este trabajo es explorar las relaciones entre profundidad del agua subterránea y variables medidas en el suelo para evaluar el desarrollo de metodologías de monitoreo alternativas, usando el Salar del Huasco como caso de estudio.

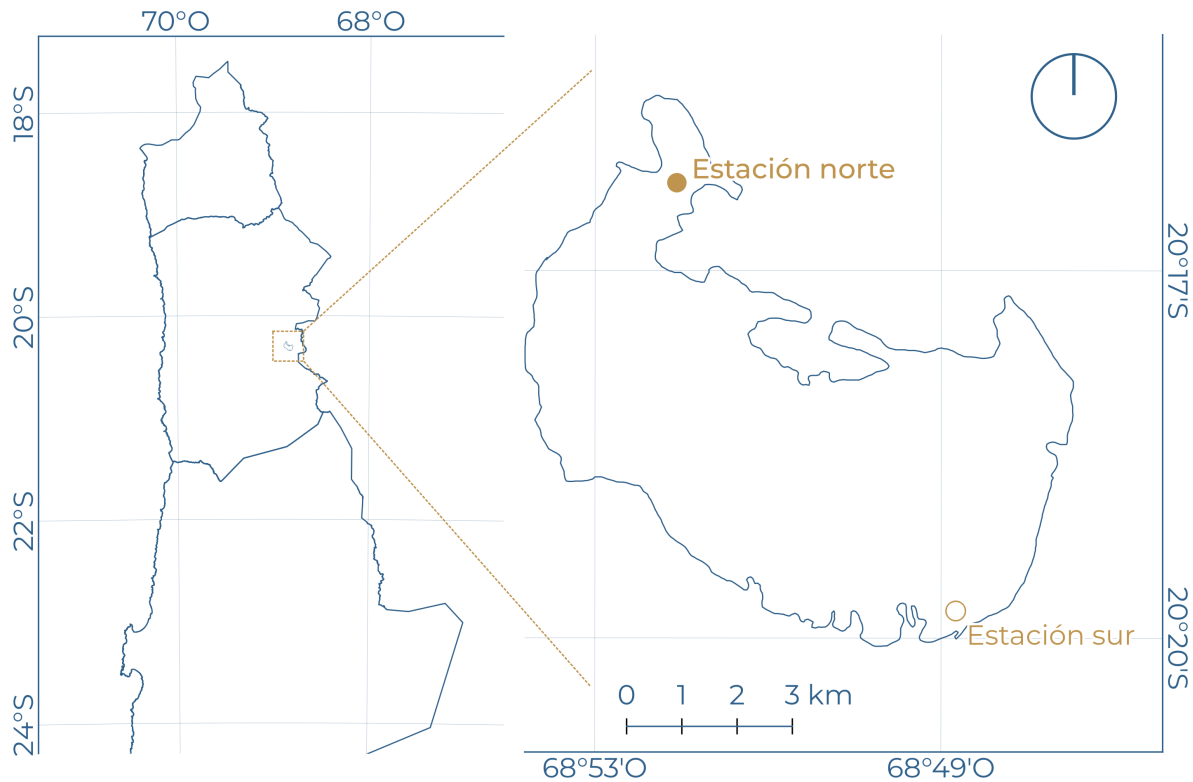


Figure 1: Localización del Salar del Huasco y sus estaciones de monitoreo

En la actualidad, se cuenta con dos estaciones de monitoreo en el sitio: norte y sur. En este trabajo se emplearán los datos recolectados en la estación norte, donde se cuenta con un piezómetro y tres sensores de suelo ubicados a diferentes profundidades (15, 30 y 48 cm). El piezómetro reporta la profundidad del nivel freático (m), mientras que los sensores de suelo, datos de contenido de agua (m^3/m^3) y temperatura ($^{\circ}\text{C}$).

Análisis exploratorio

Los datos obtenidos desde los piezómetros y sensores de suelo fueron formateados, se le removieron datos anómalos y se agregaron diariamente. Los scripts correspondientes están disponibles [aquí](#). Los datos crudos y procesados están disponibles [aquí](#).

En la siguiente celda de código se realiza el procesamiento previo a la visualización, consistente en la lectura y formateo de los datos diarios.

```
import pandas as pd
import seaborn as sns
```

```

import matplotlib.pyplot as plt
import matplotlib.dates as mdates

# Cargar Datos
piezo_df = pd.read_csv(
    '../data/processed/03_daily/piezo-data_SDH1PS01_daily.csv',
    parse_dates=['Timestamps'], index_col='Timestamps',
    usecols=['Timestamps', 'depth_m']
)

soil_df = pd.read_csv(
    '../data/processed/03_daily/soil-data_z6-25818_daily.csv',
    parse_dates=['Timestamps'], index_col='Timestamps',
    usecols=['Timestamps',
              'TEROS12_15cm_water-content_m3/m3', 'TEROS12_30cm_water-content_m3/m3', 'TEROS12_48cm_water-content_m3/m3',
              'TEROS12_15cm_soil-temperature_degree_C', 'TEROS12_30cm_soil-temperature_degree_C', 'TEROS12_48cm_soil-temperature_degree_C']
)

# Diccionarios de mapeo para nombres más limpios y extracción de metadatos
# Formato: Sensor_Profundidad_Variable
soil_rename_map = {
    'TEROS12_15cm_water-content_m3/m3': 'Suelo_15cm_Contenido-agua',
    'TEROS12_30cm_water-content_m3/m3': 'Suelo_30cm_Contenido-agua',
    'TEROS12_48cm_water-content_m3/m3': 'Suelo_48cm_Contenido-agua',
    'TEROS12_15cm_soil-temperature_degree_C': 'Suelo_15cm_Temperatura',
    'TEROS12_30cm_soil-temperature_degree_C': 'Suelo_30cm_Temperatura',
    'TEROS12_48cm_soil-temperature_degree_C': 'Suelo_48cm_Temperatura'
}

# Renombrar columnas
soil_df = soil_df.rename(columns=soil_rename_map)
piezo_df = piezo_df.rename(columns={'depth_m': 'Piezometro_NA_Profundidad-nivel-freatico'})

# Unir datasets
df_merged = pd.merge(piezo_df, soil_df, left_index=True, right_index=True, how='inner')

# Transformar a formato largo
df_long = df_merged.reset_index().melt(id_vars='Timestamps', var_name='metadata', value_name='value')

# Extraer metadatos dividiendo el string por guiones bajos (estructura: Sensor_Profundidad_Variable)
metadata_split = df_long['metadata'].str.split('_', expand=True)
df_long['Sensor'] = metadata_split[0] # Ej: Suelo, Piezometro

```

```

df_long['Profundidad'] = metadata_split[1] # Ej: 15cm, 30cm, NA
df_long['Variable'] = metadata_split[2] # Ej: Contenido-agua, Temperatura

# Generacion de un espacio entre el numero y la unidad para etiquetas
df_long['Profundidad_label'] = df_long['Profundidad'].str.replace('cm', ' cm').replace('NA', 'NA cm')

# Definir etiquetas limpias para los gráficos
variable_labels = {
    'Profundidad-nivel-freatico': 'Prof. del nivel freático (m)',
    'Contenido-agua': 'Contenido de agua (m³/m³)',
    'Temperatura': 'Temperatura (°C)'
}

# Mapeamos para usar en el facet grid
df_long['Variable_label'] = df_long['Variable'].map(variable_labels)

# Orden específico para las profundidades en la leyenda
soil_depth_order = ['15 cm', '30 cm', '48 cm']
full_depth_order = ['15 cm', '30 cm', '48 cm', 'NA']

```

En primer lugar, se graficaron las series de tiempo de profundidad del nivel freático, así como de contenido de agua y temperatura del suelo a las tres profundidades disponibles.

```

# Configuración de estilo
sns.set_theme(style="ticks", context="notebook", font_scale=1)
plt.rcParams['axes.grid'] = True
plt.rcParams['grid.alpha'] = 0.3
plt.rcParams['figure.figsize'] = (12, 8)

# Configuración de paleta de colores
viridis_colors = sns.color_palette("viridis_r", n_colors=3)
palette_dict = {
    '15 cm': viridis_colors[0], # Más claro (verde claro)
    '30 cm': viridis_colors[1], # Medio (verde azulado)
    '48 cm': viridis_colors[2], # Más oscuro (morado)
    'NA': 'black' # Negro
}

# Generar la figura base usando relplot
g = sns.relplot(
    data=df_long,
    x='Timestamps',
    y='Valor',

```

```

row='Variable_label',      # Crea una fila por cada variable
hue='Profundidad_label',   # Diferencia colores según profundidad
kind='line',
palette=palette_dict,      # Uso de diccionario de colores personalizado
hue_order=full_depth_order,
height=3,
aspect=3.5,
facet_kws={'sharey': False, 'sharex': True} # Eje X compartido, Eje Y independiente
)

# Personalización de ejes y paneles
g.set_titles("") # Eliminar los títulos sobre cada panel
g.set_xlabel("") # Eliminar la etiqueta del eje X

# Iterar sobre cada panel (subplot) para aplicar estilos específicos
for i, (ax, title) in enumerate(zip(g.axes.flat, g.row_names)):
    # Etiqueta del eje Y: Texto normal y alineado
    ax.set_ylabel(title, fontweight='normal')

    # Configuración específica para el primer panel (Nivel Freático)
    if 'Prof. del nivel freático (m)' in title:
        ax.invert_yaxis() # Invertir eje

        # Dibujar líneas de referencia horizontales para la ubicación de sensores
        sensor_depths_m = [0.15, 0.30, 0.48]
        sensor_labels = ['15 cm', '30 cm', '48 cm']

        for d_m, lbl in zip(sensor_depths_m, sensor_labels):
            # Línea punteada del color correspondiente al sensor
            ax.axhline(y=d_m, color=palette_dict[lbl], linestyle='--', linewidth=1, alpha=0.8)

    # Formato de fecha en el eje X
    date_fmt = mdates.DateFormatter('%d %b %Y')
    ax.xaxis.set_major_formatter(date_fmt)

# Personalización de la leyenda
# Extraer los objetos (handles) y etiquetas actuales del primer gráfico
handles, labels = g.axes[0,0].get_legend_handles_labels()

# Filtrar la lista para excluir 'NA' (Nivel freático), ya que es la línea negra
clean_handles = [h for h, l in zip(handles, labels) if l != 'NA']

```

```

clean_labels = [l for l in labels if l != 'NA']

# Eliminar la leyenda por defecto a la derecha
g._legend.remove()

# Crear una nueva leyenda personalizada
g.fig.legend(
    clean_handles,
    clean_labels,
    loc='lower center',
    bbox_to_anchor=(0.5, 0),
    ncol=3, # Disposición horizontal
    title="Profundidad de los sensores de suelo",
    frameon=False
)

# Ajustes finales
# Título principal de la figura
g.fig.suptitle("Evolución de variables piezométricas y edáficas", x=0.5, y=0.98, fontsize=14)
# Ajustes de espaciado
plt.tight_layout()
plt.subplots_adjust(top=0.93, bottom=0.11) # Ajusta espacio disponible para subplots

plt.show()

```

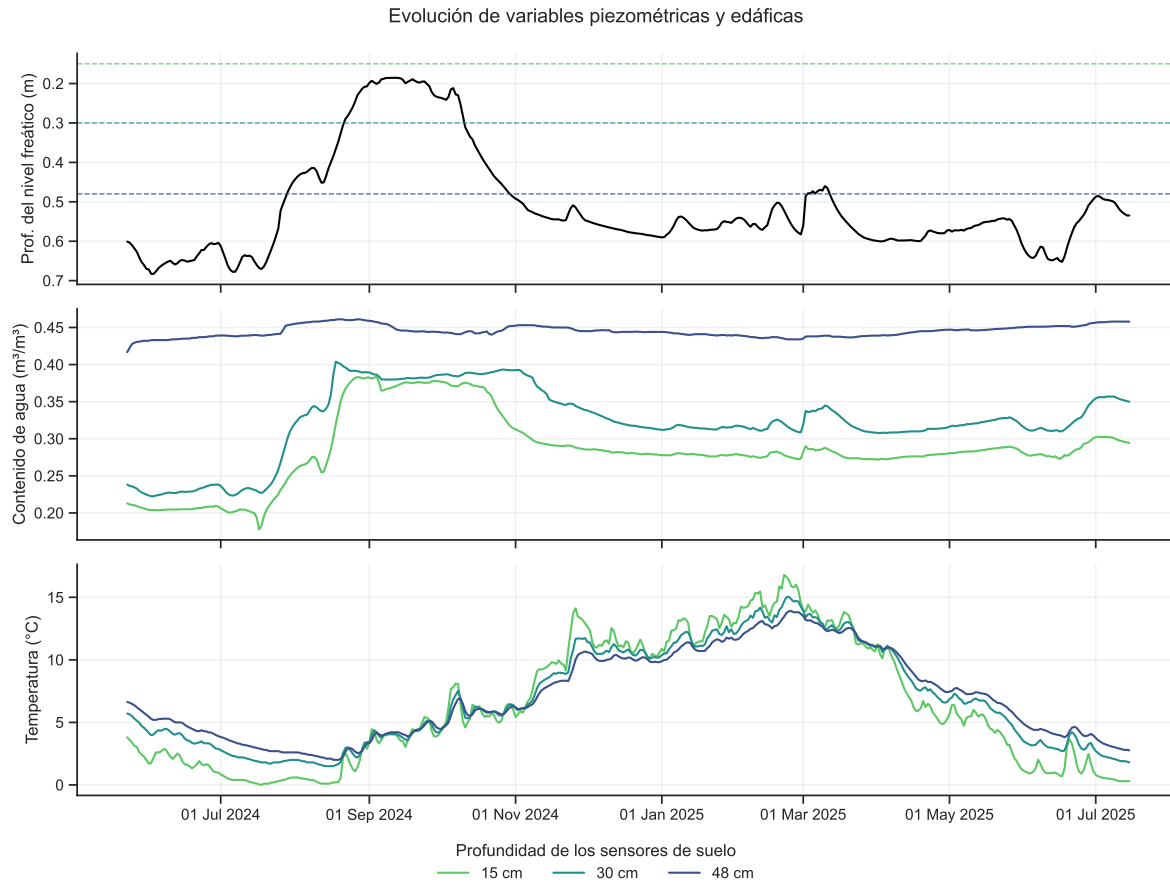


Figure 2: Series temporales de profundidad del nivel freático, contenido de agua y temperatura del suelo

Luego, se graficó la relación entre profundidad del nivel freático y las variables de suelo mediante gráficos de dispersión.

```
# Preparación de datos para scatter plot
# Filtrar los datos de suelo
df_soil = df_long[df_long['Sensor'] == 'Suelo'].copy()

# Extraer la serie de nivel freático (NA) y convertir en una columna
piezo_col = df_long[df_long['Profundidad_label'] == 'NA'].set_index('Timestamps')['Valor']

# Unir por fecha
df_scatter = df_soil.merge(piezo_col.rename('GWD'), left_on='Timestamps', right_index=True)
```

```

# Generar la figura base usando relplot scatter
g2 = sns.relplot(
    data=df_scatter,
    x='Valor',                # Eje X: Variables de suelo
    y='GWD',                  # Eje Y: Nivel freático
    row='Profundidad_label',  # Filas: Profundidad (15, 30, 48)
    col='Variable_label',     # Columnas: Variable (Contenido-agua, Temperatura)
    hue='Profundidad_label',
    kind='scatter',
    palette=palette_dict,
    row_order=soil_depth_order, # 15, 30, 48
    hue_order=soil_depth_order,
    height=3,
    aspect=1.2,
    facet_kws={'sharey': True, 'sharex': 'col'},
    alpha=0.7,
    s=30
)

# Personalización de ejes y paneles
# Eliminar los títulos automáticos de los subplots
g2.set_titles("")

# Inversión del eje Y
# Calcular min y max de los datos para establecer los límites manualmente
y_min = df_scatter['GWD'].min()
y_max = df_scatter['GWD'].max()
padding = (y_max - y_min) * 0.05 # Agregar un 5% de espacio
g2.set(ylim=(y_max + padding, y_min - padding))

# Configurar etiquetas del eje X en la fila inferior
g2.set_xlabel("") # Limpiar etiquetas automáticas
g2.axes[-1, 0].set_xlabel("Contenido de agua (m³/m³)")
g2.axes[-1, 1].set_xlabel("Temperatura (°C)")

# Etiqueta Global del Eje Y
g2.set_ylabel("") # Borrar etiquetas individuales automáticas
g2.fig.supylabel(
    "Profundidad del nivel freático (m)",
    x=0.04,
    size=plt.rcParams['axes.labelsize'], # Igualar tamaño al eje X
)

```



```

# Personalización de la leyenda
# Extraer handles/labels del primer panel
handles, labels = g2.axes[0,0].get_legend_handles_labels()

# Eliminar la leyenda automática
g2._legend.remove()

# Crear leyenda personalizada
g2.fig.legend(
    handles=handles,
    labels=labels,
    loc='lower center',
    bbox_to_anchor=(0.5, 0),
    ncol=3,
    title="Profundidad de los sensores de suelo",
    frameon=False
)

# Ajustes finales
# Título principal
g2.fig.suptitle("Relación entre variables de suelo y profundidad del nivel freático", y=0.98)

# Ajustes de espaciado
plt.tight_layout()
plt.subplots_adjust(top=0.94, bottom=0.13, left=0.11) # Ajusta espacio disponible para subplots

plt.show()

```

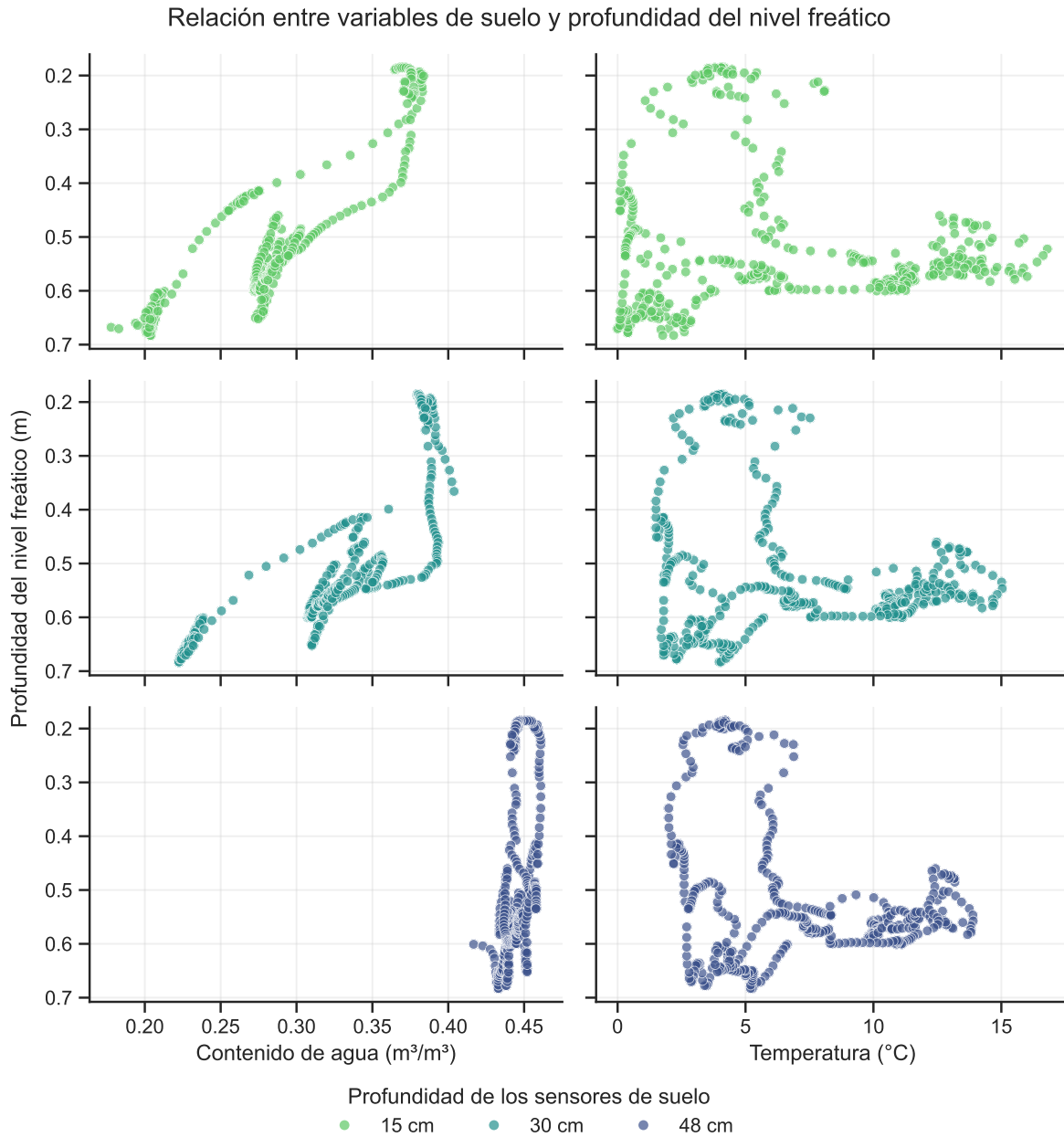


Figure 3: Relaciones entre profundidad del nivel freático y variables edáficas

- Tabla con resultados de tests para evaluar estacionariedad de los datos (ADF, KPSS) antes y después del detrending.
- Gráficas de series temporales sin tendencia: similar a gráfica 1.
- Gráficas de coeficientes de correlación anuales: dos facet (uno por variable: contenido de agua y temperatura), cada uno de tres filas (una por profundidad). Los gráficos

- comparten el eje X (número de lags).
- Gráficas de coeficientes de correlación para una temporada específica (dependiente de la visualización preliminar).

Desafíos

Desarrollar una conclusión a partir de los resultados iniciales y cuáles serían próximos pasos a desarrollar para continuar con la investigación:

- Exploración de correlaciones en diferentes momentos temporales
- Exploración de correlaciones con datos de otros sitios
- Exploración de correlaciones con otro tipo de datos (meteorológicos, flujos de calor)