# BIG DATA

## Presentation

# GROUNDWATER ANALYSIS IN CALIFORNIA

Angeline Mary Marchella
Darian Elbert
Pattrick Ritter
Thomas Dante Wunan
Yosua Raffel Istianto

# Objective

Groundwater Analysis in California

## Exploratory Data Analysis

We will explore the available data and some additional data to find some relation between them

01

02

## Regression

We want to perform a model training to predict the groundwater level changes (diff_gwe) based on certain features (preciptation, temperature, and county information)

# Starting Spark Session

Groundwater Analysis in California

```
!pip install -q pyspark
```

```
[ ]  from pyspark.sql import SparkSession
     from pyspark.ml.feature import VectorAssembler
     from pyspark.sql.functions import avg
     from pyspark.sql.functions import col

     spark = SparkSession \
         .builder \
         .appName("Big Data Final Project") \
         .config("spark.executor.memory", "8g") \
         .config("spark.driver.memory", "4g") \
         .getOrCreate()
```

# Importing Dataset

Groundwater Analysis in California

```
[ ]  # 1. Install 'kaggle' library
     !pip install -q kaggle

     # 2. Create a directory named '.kaggle' at the root folder of Google Colab
     !mkdir -p ~/.kaggle
```

```
[ ]  # creating kaggle.json file that store the kaggle API
     import json

     data = {"username":"angelinemarym","key":"[redacted]"}
     jd = json.dumps(data)

     with open('kaggle.json', 'w') as f:
       f.write(jd)
```

```
[ ]  # 3. Copy the "kaggle.json" file to the current storage and change its modifier
     !cp kaggle.json ~/.kaggle/
     !chmod 600 /root/.kaggle/kaggle.json

     # 4. Download the dataset and other important stuff from kaggle
     ! kaggle datasets download angelinemarym/periodic-groundwater-level-measurements

     # 5. Unzip the downloaded file to be processed
     ! unzip periodic-groundwater-level-measurements.zip
```

**Obtaining kaggle API**

Settings > Account menu >
API section > Create New Token

**Dataset:**

- Periodic Water Measurement in California
- Collection Stations dataset
- California's County & City boundaries
- Water Quality Measurement
- Periodic Temperature
- Periodic Percipitation

# Preprocessing

Groundwater Analysis in California

```
df = df.join(miss, df['site_code'] == miss['site_code'], 'left_anti')
```

```
df = df.withColumn("date", to_date(col("msmt_date")))
```

```
[ ]  matching_df = df.join(df2,       (df["latitude"] == df2["GM_LATITUDE"]) &
                                      (df["longitude"] == df2["GM_LONGITUDE"]), "inner")
```

```
[ ]  df = df.filter(df['county_name'] != 'Klamath, OR')
```

```
df = df.withColumn('year', year(col('date')))
```

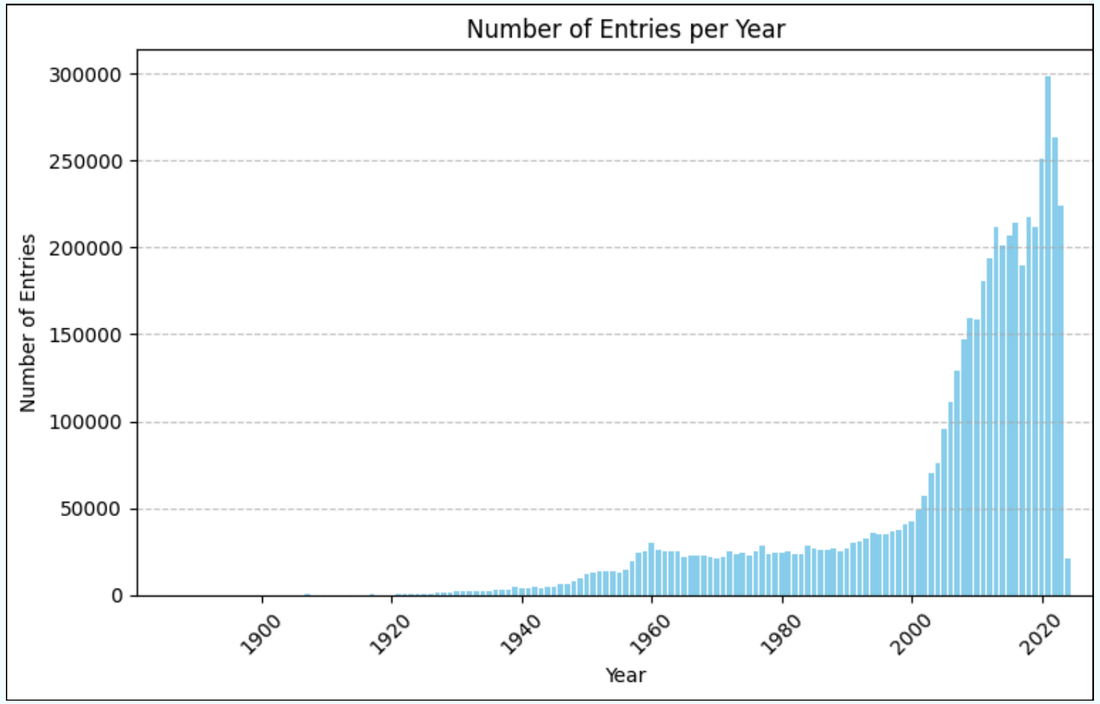
```
[134] x = x.dropna()
```

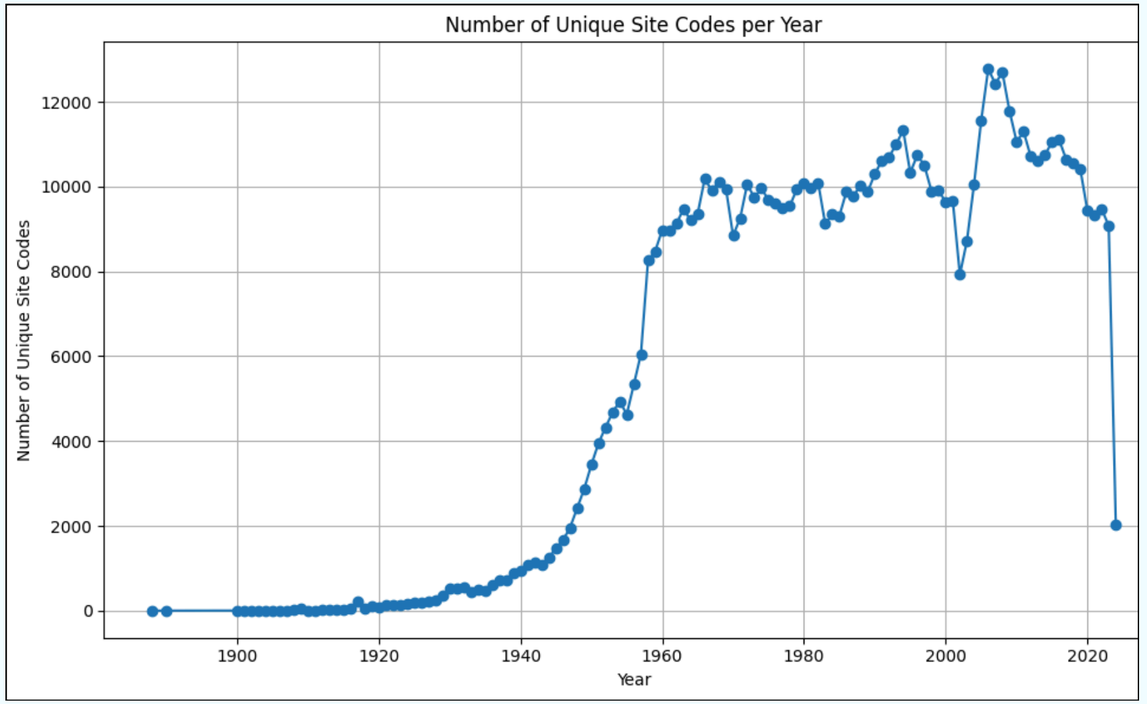
kaggle

# Exploratory Data Analysis (EDA)

# Number of well entires per year

Groundwater Analysis in California



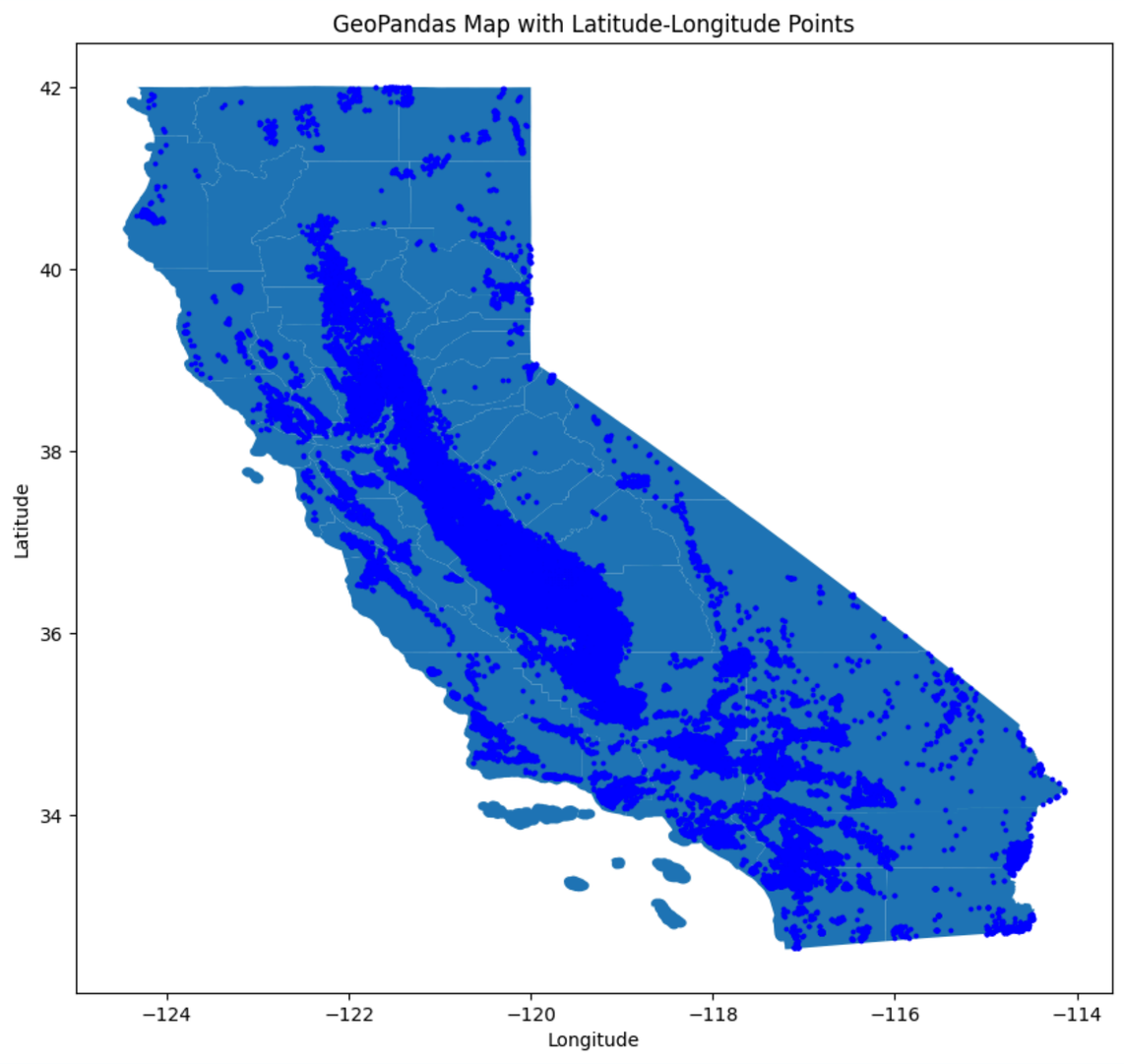
# Number of well entires per year

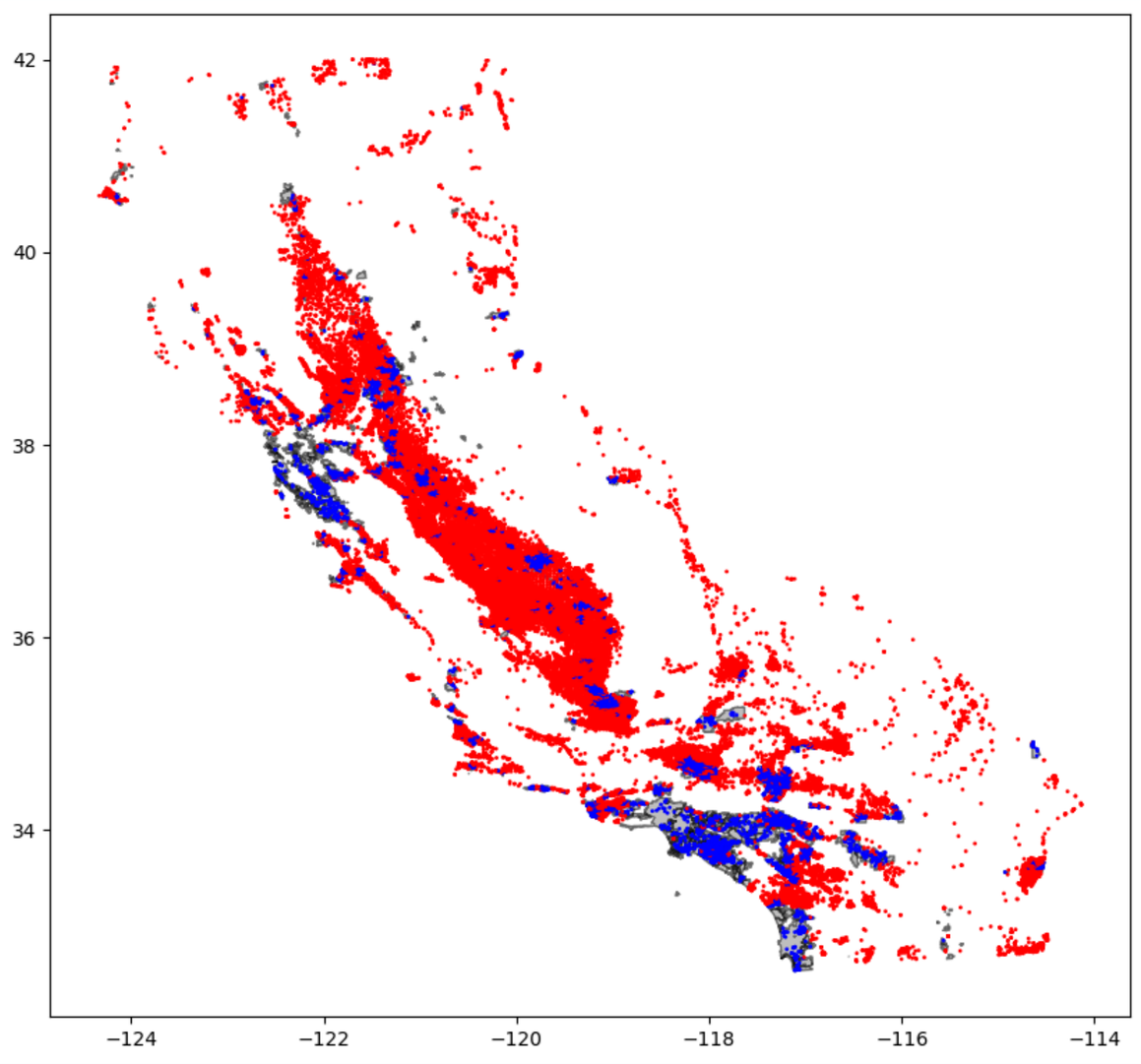
Groundwater Analysis in California

| | county_name | unique_site_count_per_county |
|---|---|---|
| 0 | Plumas | 152 |
| 1 | Kings | 1475 |
| 2 | Marin | 7 |
| 3 | Inyo | 327 |
| 4 | Sonoma | 457 |
| 5 | Napa | 182 |
| 6 | Madera | 1086 |
| 7 | Siskiyou | 229 |
| 8 | Ventura | 1317 |
| 9 | Orange | 613 |
| 10 | Los Angeles | 2207 |
| 11 | Sacramento | 548 |
| 12 | Lake | 168 |
| 13 | Stanislaus | 1350 |
| 14 | Calaveras | 17 |
| 15 | San Diego | 2528 |
| 16 | San Francisco | 16 |
| 17 | None | 1 |
| 18 | El Dorado | 67 |
| 19 | Santa Cruz | 323 |
| 20 | Santa Clara | 237 |
| 21 | San Joaquin | 1054 |
| 22 | Sutter | 449 |
| 23 | Del Norte | 10 |
| 24 | Nevada | 16 |
| 25 | Yuba | 262 |
| 26 | Merced | 2534 |
| 27 | Yolo | 707 |
| 28 | Tehama | 407 |
| 29 | Colusa | 245 |
| 30 | Amador | 23 |
| 31 | San Benito | 183 |
| 32 | Contra Costa | 132 |
| 33 | Sierra | 62 |
| 34 | Fresno | 6141 |
| 35 | Mendocino | 156 |
| 36 | San Mateo | 43 |
| 37 | Tuolumne | 8 |
| 38 | Single Well | 1 |
| 39 | Modoc | 154 |
| 40 | Butte | 335 |
| 41 | Mono | 140 |
| 42 | Kern | 5290 |
| 43 | Solano | 378 |
| 44 | Glenn | 372 |
| 45 | Klamath, OR | 8 |
| 46 | Lassen | 154 |
| 47 | Humboldt | 138 |
| 48 | Alpine | 34 |
| 49 | San Luis Obispo | 543 |
| 50 | Riverside | 2776 |
| 51 | Imperial | 475 |
| 52 | Mariposa | 8 |
| 53 | Santa Barbara | 729 |
| 54 | Monterey | 368 |
| 55 | Shasta | 101 |
| 56 | San Bernardino | 5904 |
| 57 | Alameda | 510 |
| 58 | Tulare | 2279 |
| 59 | Placer | 187 |

# Location of Wells

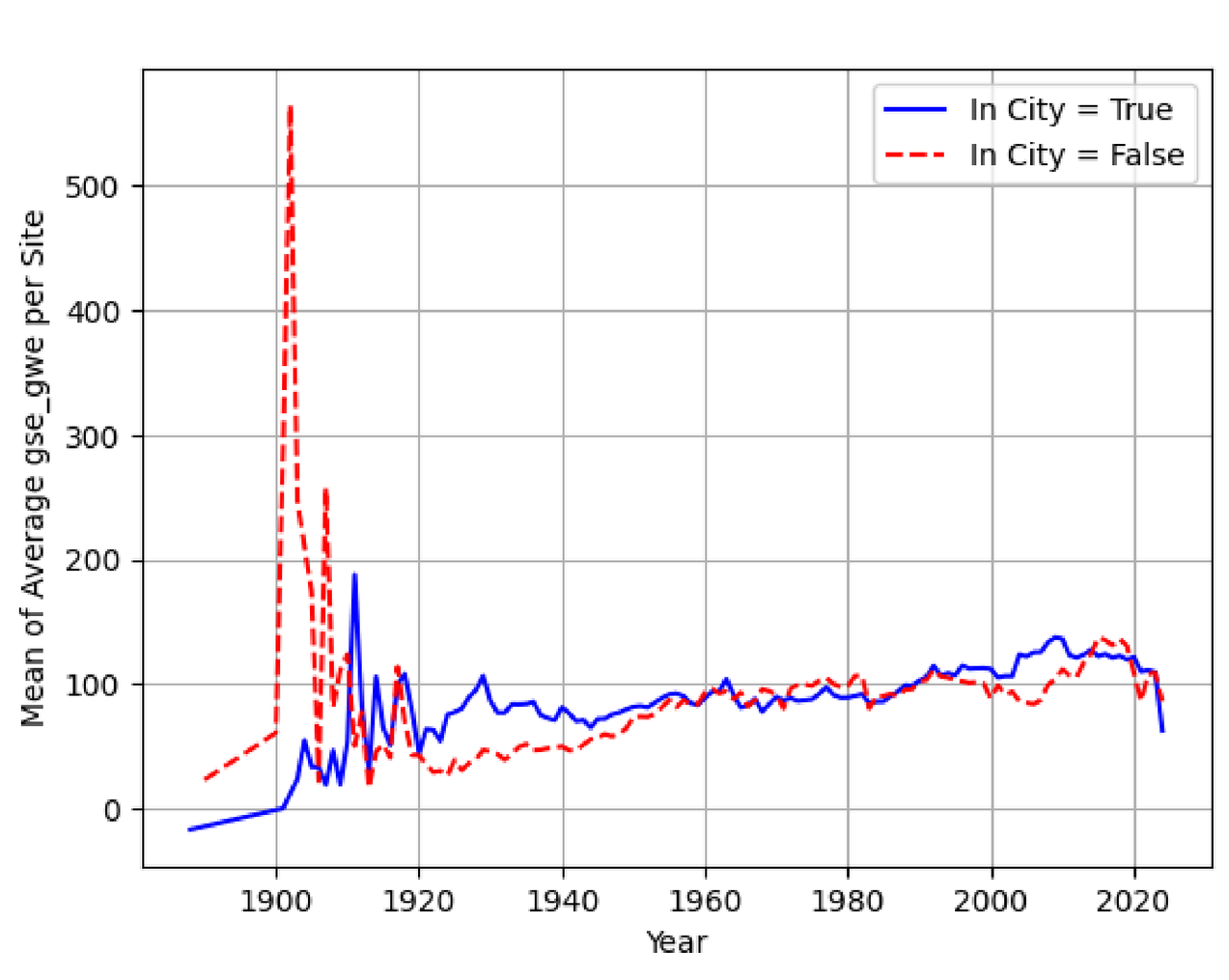
Groundwater Analysis in California



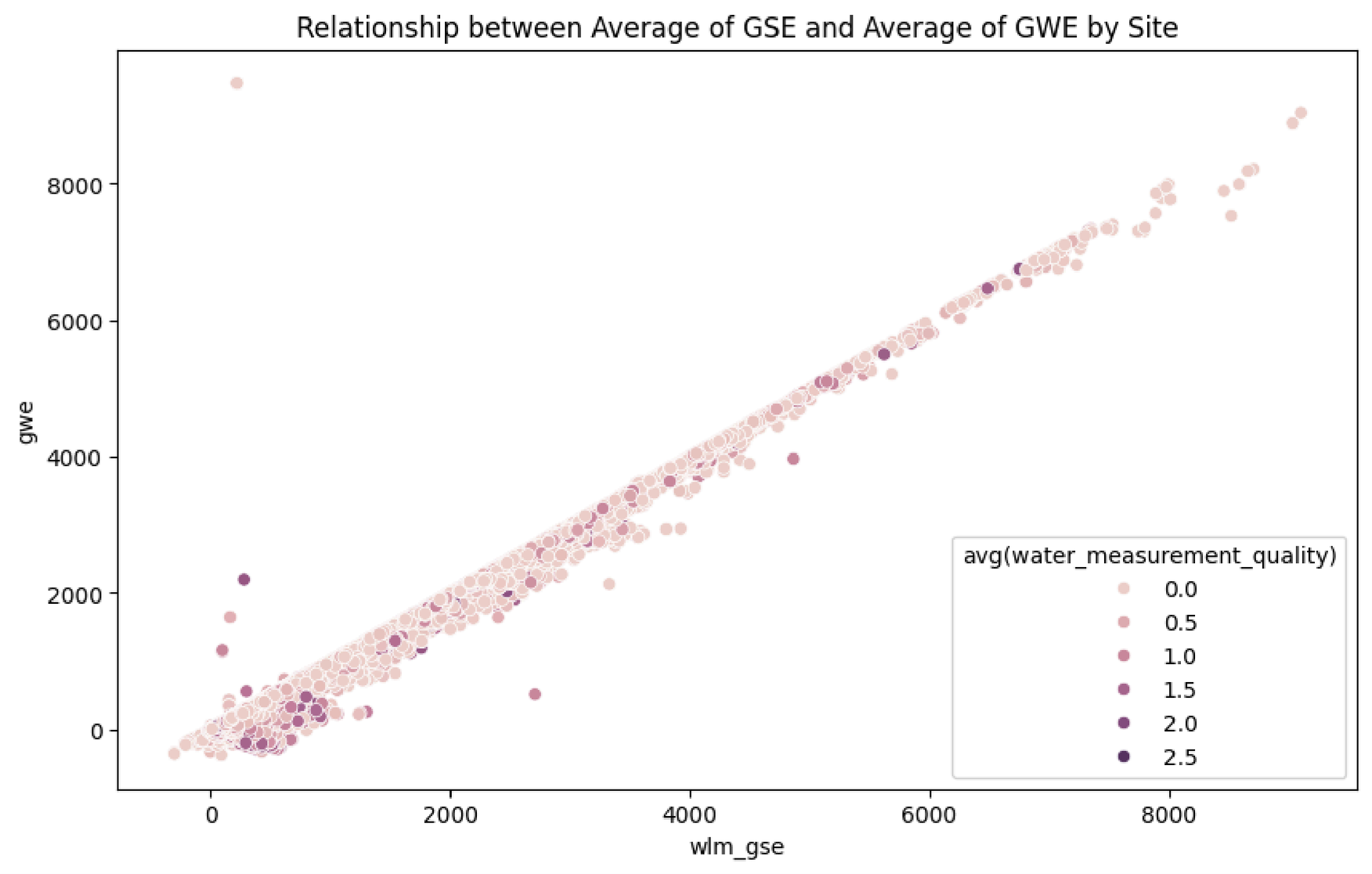
# GWE City vs Non City

Groundwater Analysis in California



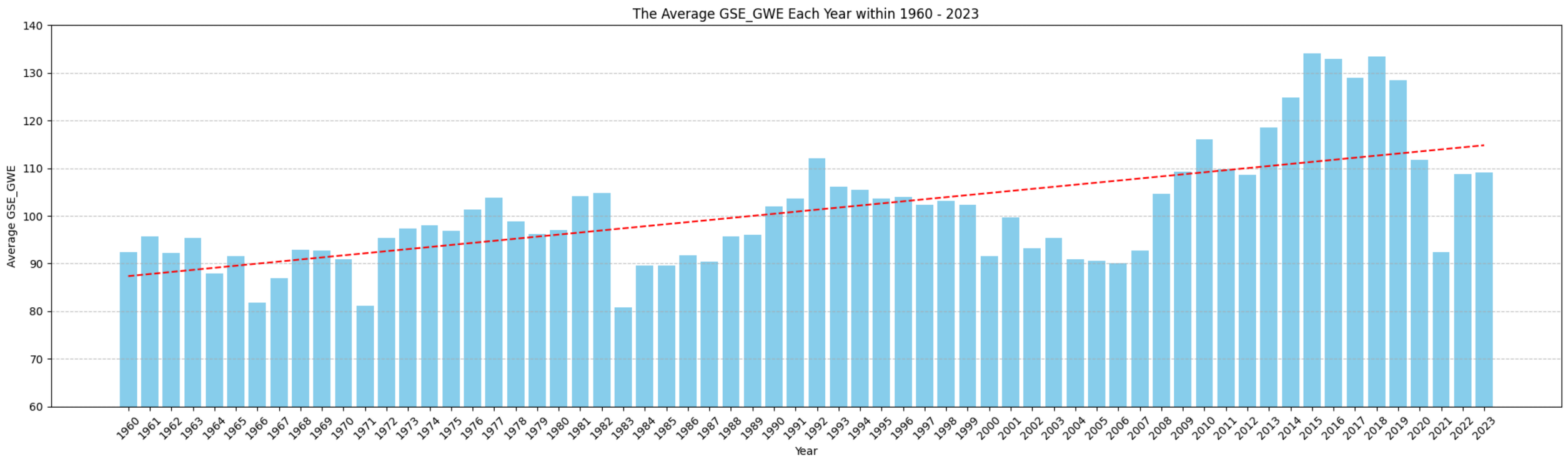
# GSE & GWE Analysis (Correlation)

Groundwater Analysis in California



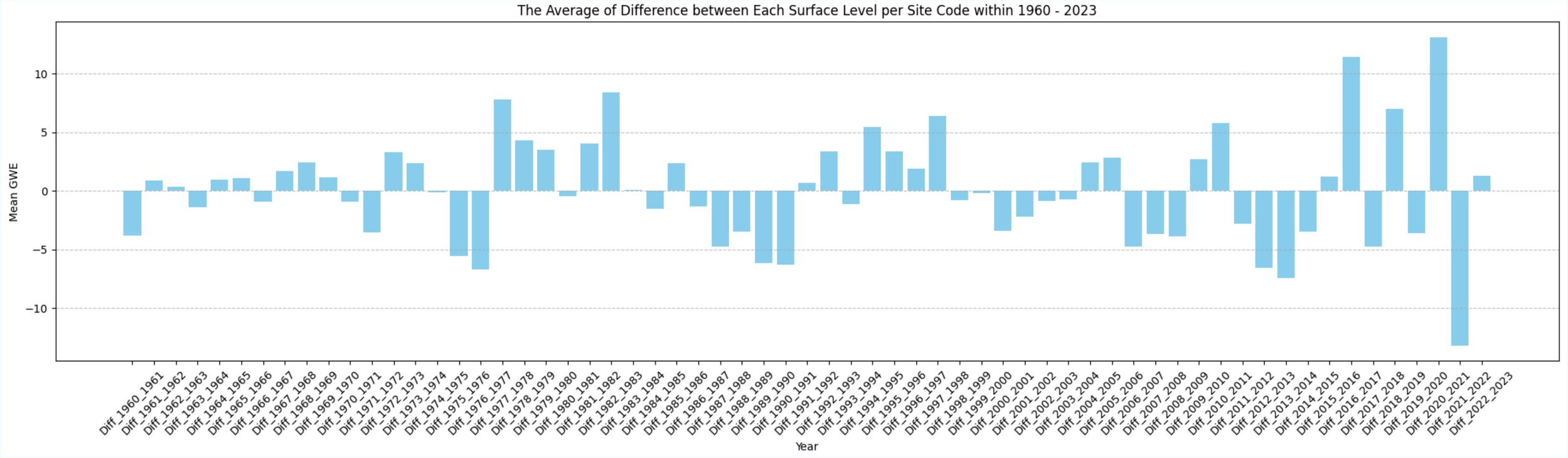
# GSE & GWE Analysis (Avg per Year)

Groundwater Analysis in California



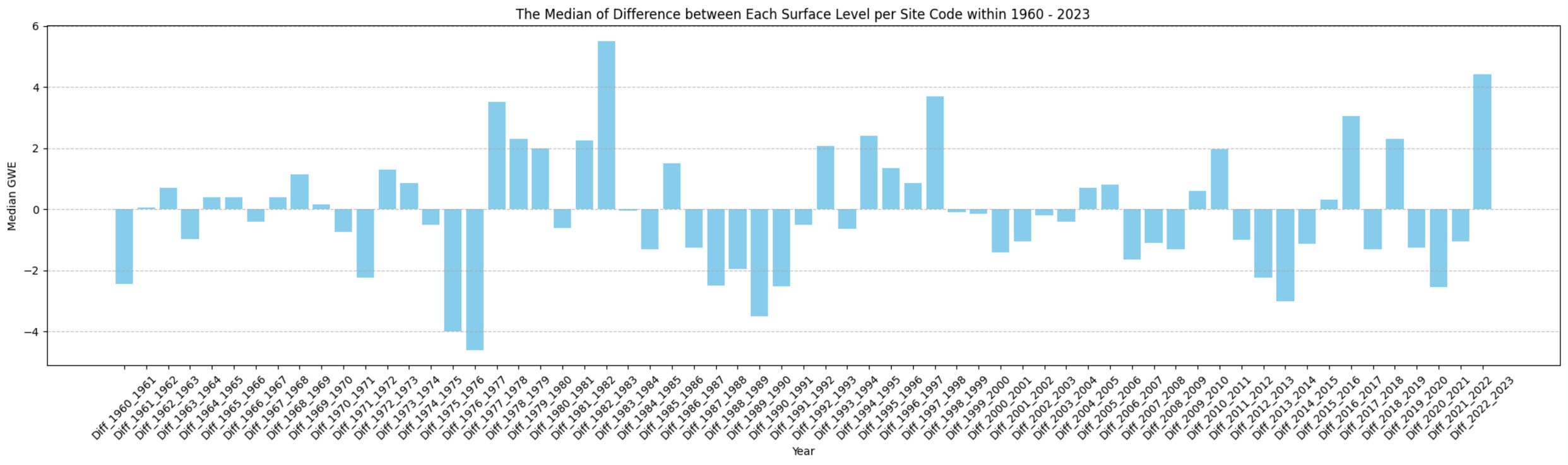
# Surface Level Difference (Site Code | Average)

Groundwater Analysis in California



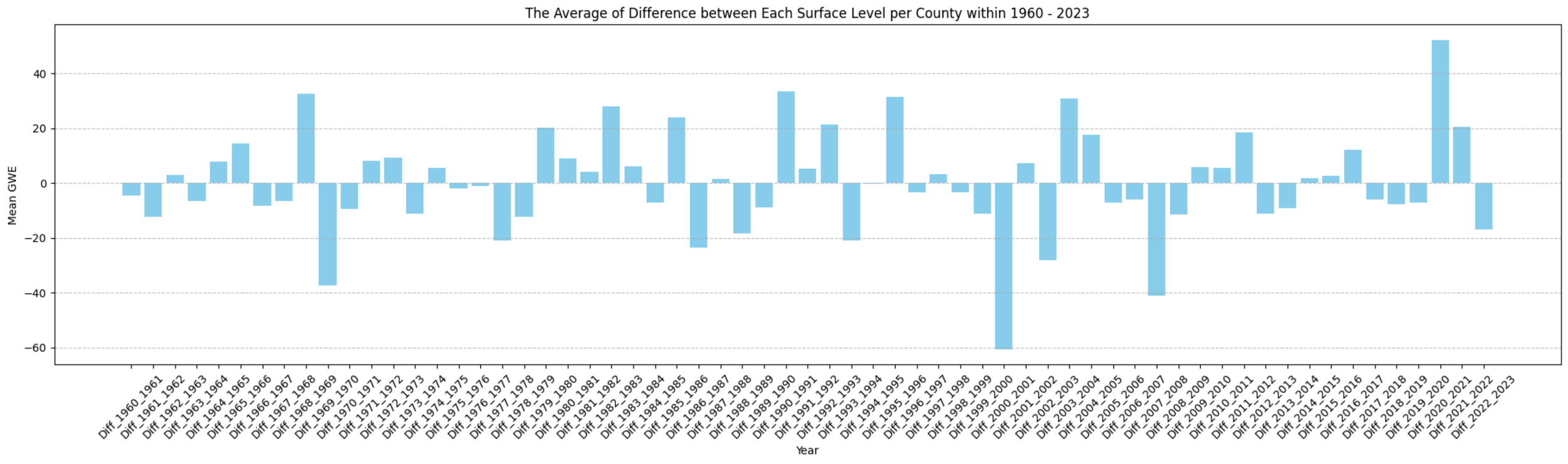
# Surface Level Difference (Site Code | Median)

Groundwater Analysis in California



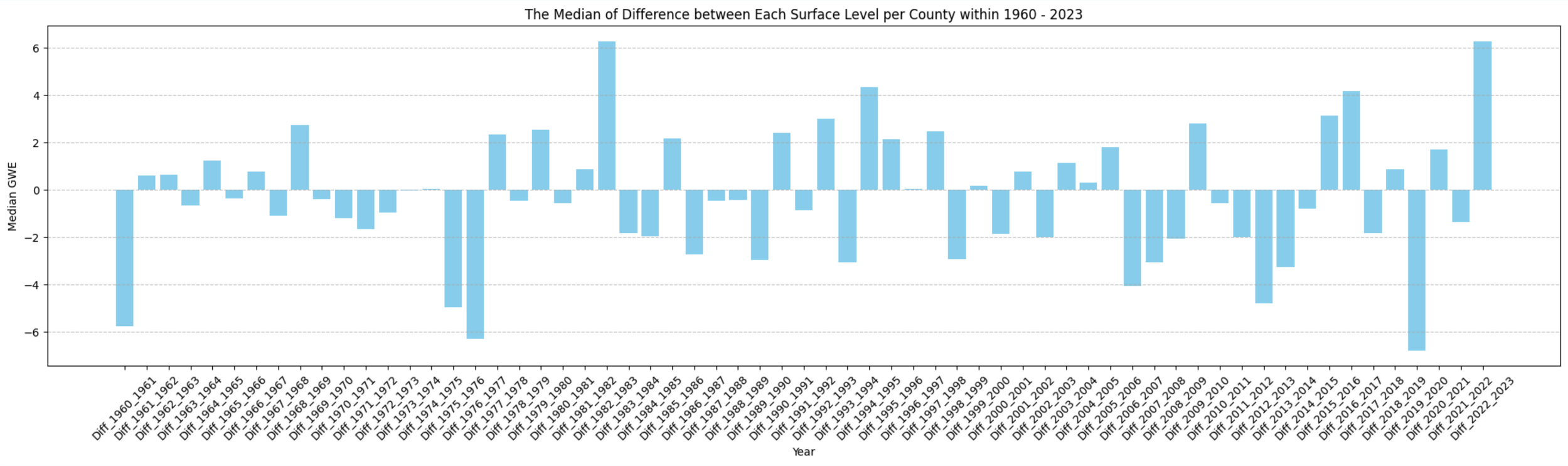
# Surface Level Difference (County/Average)

Groundwater Analysis in California



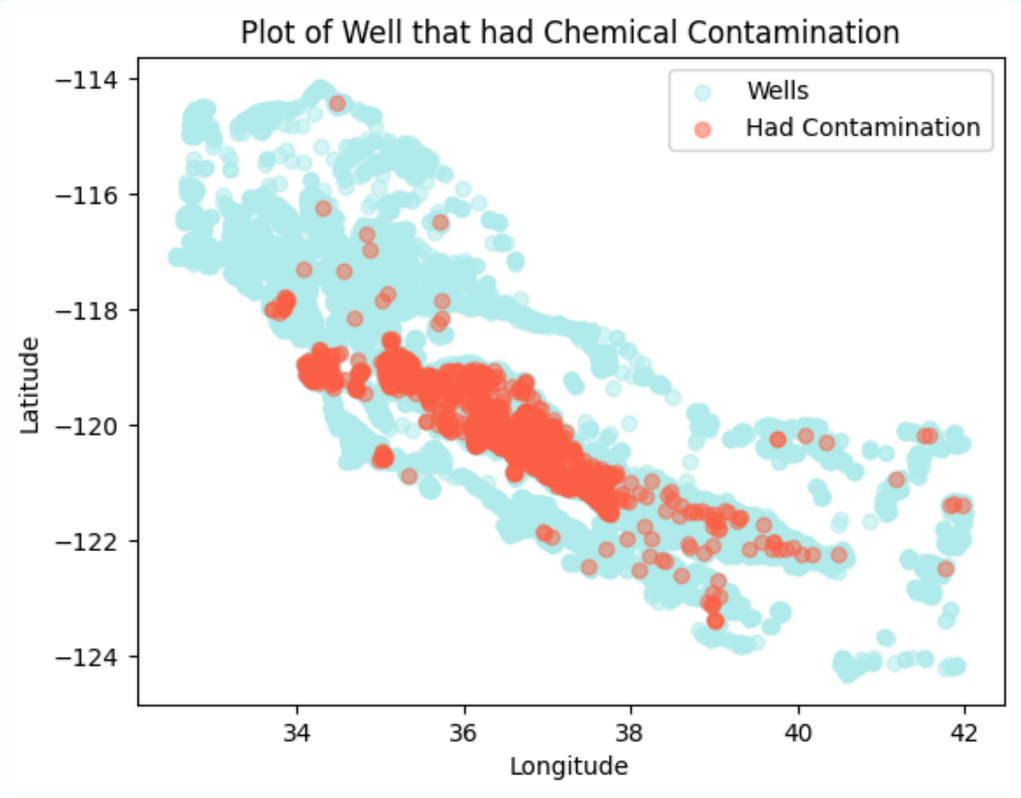
# Surface Level Difference (County/Median)

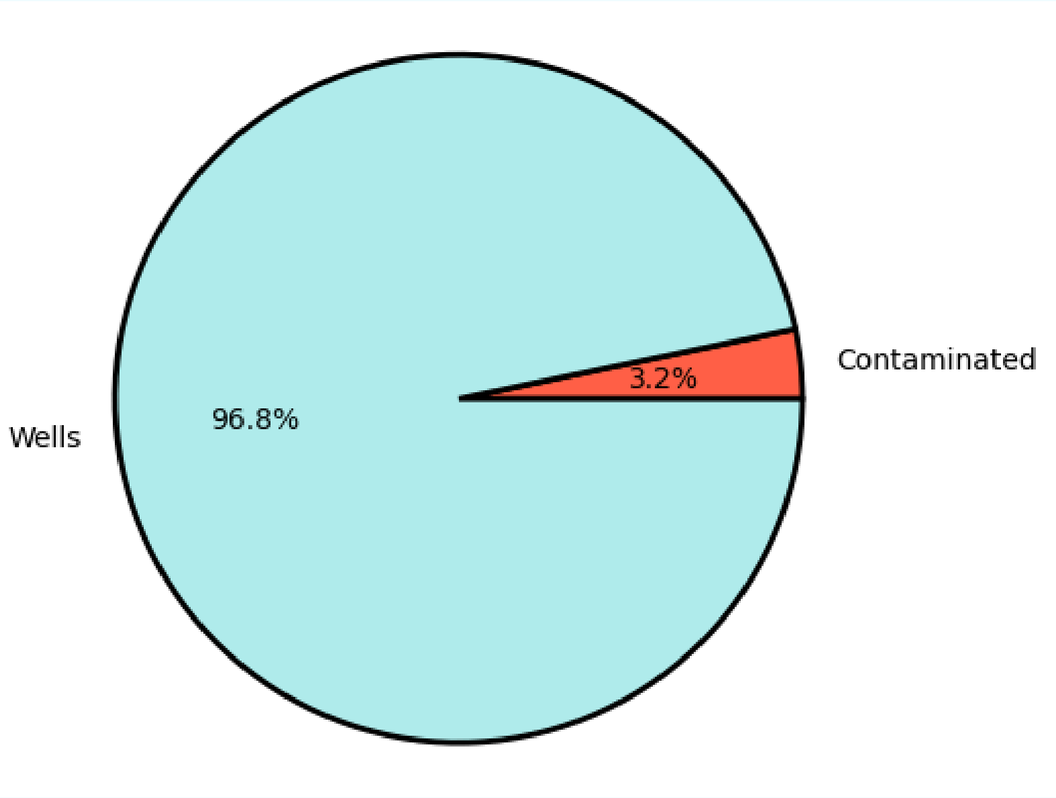
Groundwater Analysis in California



# Contamination Distribution and Plot

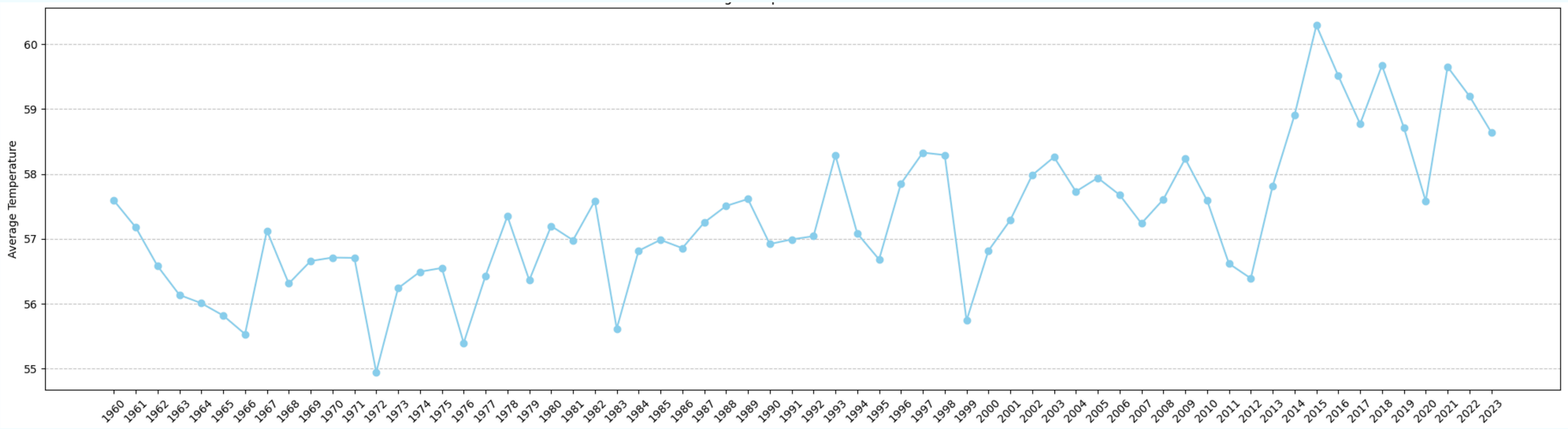
Groundwater Analysis in California



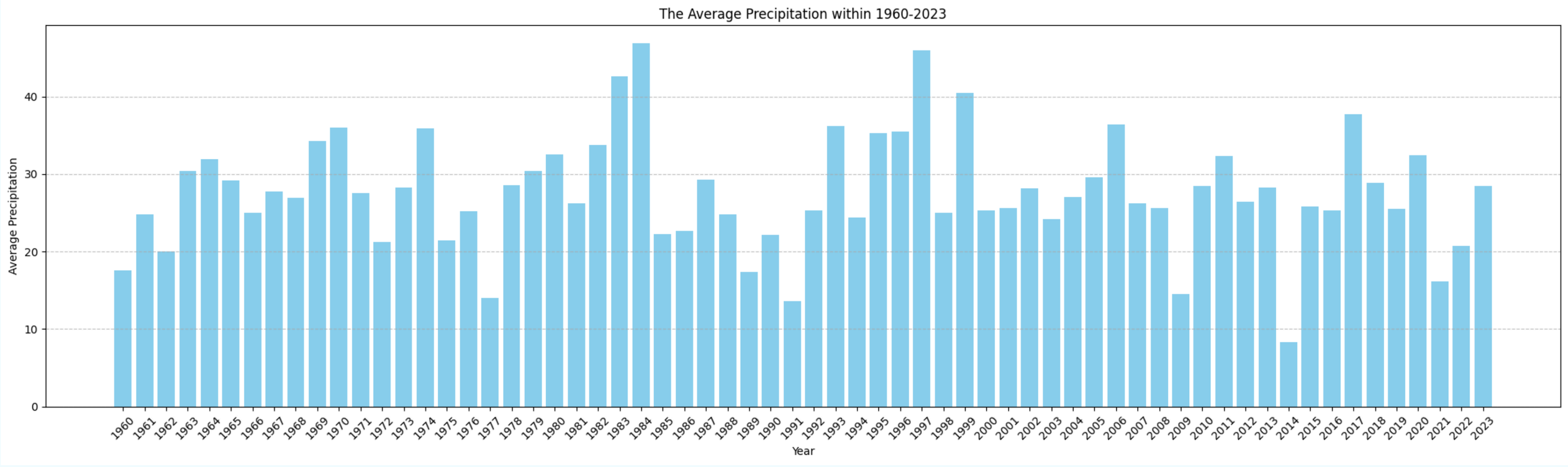
# Average Temperature

Groundwater Analysis in California



# Average Precipitation

Groundwater Analysis in California



# Predicting Groundwater level changes using Regression

# Preprocessing & Merging Dataset

Groundwater Analysis in California

```
precip = spark.read.csv('precipitation2.csv', header = True, inferSchema = True)
temp = spark.read.csv('avg_temp.csv', header = True, inferSchema = True)
names = spark.read.csv('County_names.csv', header = True, inferSchema = True)
```

```
merged_temp_precip_df = precip.join(temp,(precip["Precipitation_Year"] == temp["Temperature_Year"]) &
                                    (precip["Precipitation_County_Idx"] == temp["Temperature_County_Idx"]), "inner")
```

```
merged_frst = merged_temp_precip_df.drop("Precipitation_Year")
merged_frst = merged_frst.drop("Precipitation_County")
merged_frst = merged_frst.drop("Precipitation_County_Idx")
```

```
stacked_df.dtypes
```

```
[('County', 'string'),
 ('year', 'int'),
 ('diff_gwe', 'double'),
 ('County_Idx', 'double')]
```

```
merged_scnd = stacked_df.alias("stacked_df").join(merged_frst.alias("merged_frst"),
    (stacked_df["year"] == merged_frst["Temperature_Year"]) &
    (stacked_df["County_Idx"] == merged_frst["Temperature_County_Idx"]), "inner")
```

```
merged_fin = merged_scnd.drop("Temperature_County_Idx")
merged_fin = merged_fin.drop("Temperature_County")
merged_fin = merged_fin.drop("Temperature_Year")
```

```
used_df = merged_fin.select(['County_Idx', 'Precipitation', 'Temperature', 'diff_gwe'])
used_df.dtypes
```

```
[('County_Idx', 'double'),
 ('Precipitation', 'double'),
 ('Temperature', 'double'),
 ('diff_gwe', 'double')]
```

# Training & Evaluate the Data

Groundwater Analysis in California

```
[ ]  1 vectorAssembler = VectorAssembler(inputCols = ['County_Idx','Precipitation', 'Temperature'], outputCol = 'features')
     2 x = vectorAssembler.transform(used_df)
```

```
[ ]  1 x = x.select(['features', 'diff_gwe'])
```

```
[ ]  1 x = x.dropna()
```

```
[ ]  1 train, test = x.randomSplit([0.7, 0.3], seed = 42)
```

```
[ ]  rf = RandomForestRegressor(featuresCol='features', labelCol='diff_gwe', maxBins=64)
     model = rf.fit(train)
```

```
[ ]  pred = model.transform(test)
```

```
[ ]  eval = RegressionEvaluator(labelCol='diff_gwe', predictionCol='prediction', metricName='rmse')
     rmse = eval.evaluate(pred)
```

```
[ ]  rmse

     85.58940622285324
```

# Evaluation with different models

Groundwater Analysis in California

| Model Name | RMSE | MAE |
| --- | --- | --- |
| Random Forest Regression | 122.931 | 43.164 |
| Gradient Boost Tree | 185.663 | 69.610 |
| Linear Regression | 115.929 | 39.437 |

# 感谢大家的观看

Thank You For Listening

Angeline Mary Marchella
Darian Elbert
Pattrick Ritter
Thomas Dante Wunan
Yosua Raffel Istianto