FINAL_tech-layoffs-analysis

March 16, 2025

[201]:
```
%%capture
%pip install plotly
import pandas as pd
import matplotlib.pyplot as plt
import plotly.express as px
%pip install pydantic_settings
from pydantic_settings import BaseSettings
%pip install seaborn
import seaborn as sns
import os
import plotly.io as pio
from IPython.display import IFrame
for dirname, _, filenames in os.walk('../data'):
    for filename in filenames:
        print(os.path.join(dirname, filename))
%matplotlib inline
```

This code examines the major changes in global employment trends from 2020 to 2024, primarily driven by the COVID-19 pandemic. It explores how layoffs have impacted different regions, industries, and organizations, highlighting variations in workforce reductions.

The analysis focuses on key factors such as industry-specific disruptions, the role of a company's financial health in workforce decisions, and regional disparities in layoffs. By providing a comprehensive overview of employment shifts during this period, this study aims to offer valuable insights for businesses, policymakers, and individuals navigating workforce challenges. It also sheds light on resilience, adaptation, and innovation as organizations respond to these unprecedented disruptions.

[165]:
```
import pandas as pd

# Load the two CSV files
file_1_path = "layoffs (1).csv"
file_2_path = "sorted_layoffs_data.csv"

df_1 = pd.read_csv(file_1_path)
df_2 = pd.read_csv(file_2_path)

# Standardize column names
column_mapping = {
    'company': 'Company',
```

```python
    'location': 'Location HQ',
    'industry': 'Industry',
    'total_laid_off': '# Laid Off',
    'percentage_laid_off': 'Percentage',
    'funds_raised': '$ Raised (mm)',
    'stage': 'Stage',
    'country': 'Country',
    'date': 'Date'
}

df_1.rename(columns=column_mapping, inplace=True)

df_1['Date'] = pd.to_datetime(df_1['Date'], errors='coerce')
df_2['Date'] = pd.to_datetime(df_2['Date'], errors='coerce')

df_1_filtered = df_1[df_1['Date'].dt.year.isin([2020, 2021, 2024])]
df_2_filtered = df_2[df_2['Date'].dt.year.isin([2022, 2023, 2024, 2025])]

# Convert it into same dimensions
common_columns = [
    'Company', 'Location HQ', 'Industry', '# Laid Off', 'Percentage', 'Date',␣
 ↪'Stage', 'Country', '$ Raised (mm)'
]

df_1_filtered = df_1_filtered[common_columns]
df_2_filtered = df_2_filtered[common_columns]

# Merge
final_df = pd.concat([df_1_filtered, df_2_filtered], ignore_index=True)

final_df.sort_values(by="Date", ascending=False, inplace=True)

# Save
final_file_path = "cleaned_combined_layoffs_2020_2025.csv"
final_df.to_csv(final_file_path, index=False)

print(f"Final dataset saved as '{final_file_path}'")
print(final_df.head())
```

```
Final dataset saved as 'cleaned_combined_layoffs_2020_2025.csv'
                       Company  Location HQ   Industry  # Laid Off  \
714                       D-ID     Tel Aviv         AI        22.0
715              Zonar Systems      Seattle  Logistics         NaN
716                     Wayfair       Boston     Retail       340.0
717  Hewlett Packard Enterprise  SF Bay Area   Hardware      2500.0
718                    LiveRamp  SF Bay Area  Marketing        65.0

    Percentage       Date      Stage        Country $ Raised (mm)
```

```
714         25% 2025-03-10       NaN         Israel           $48
715         NaN 2025-03-09  Acquired  United States           $50
716         NaN 2025-03-07  Post-IPO        Germany        $1,700
717          5% 2025-03-06  Post-IPO  United States        $1,400
718          5% 2025-03-05  Post-IPO  United States           $16
```

[166]:
```
data = pd.read_csv("cleaned_combined_layoffs_2020_2025.csv")
data
```

[166]:
```
                         Company     Location HQ        Industry  # Laid Off  \
0                           D-ID        Tel Aviv              AI        22.0
1                  Zonar Systems         Seattle       Logistics         NaN
2                       Wayfair           Boston          Retail       340.0
3     Hewlett Packard Enterprise     SF Bay Area        Hardware      2500.0
4                       LiveRamp     SF Bay Area       Marketing        65.0
…                            …              …              …           …
2711                     Service     Los Angeles          Travel         NaN
2712                 HopSkipDrive     Los Angeles  Transportation         8.0
2713                  Panda Squad     SF Bay Area        Consumer         6.0
2714                Tamara Mellon     Los Angeles          Retail        20.0
2715                     EasyPost  Salt Lake City       Logistics        75.0

     Percentage        Date     Stage        Country $ Raised (mm)
0           25%  2025-03-10       NaN         Israel           $48
1           NaN  2025-03-09  Acquired  United States           $50
2           NaN  2025-03-07  Post-IPO        Germany        $1,700
3            5%  2025-03-06  Post-IPO  United States        $1,400
4            5%  2025-03-05  Post-IPO  United States           $16
…          …         …        …            …            …
2711        1.0  2020-03-16      Seed  United States           5.1
2712        0.1  2020-03-13   Unknown  United States          45.0
2713       0.75  2020-03-13      Seed  United States           1.0
2714        0.4  2020-03-12  Series C  United States          90.0
2715        NaN  2020-03-11  Series A  United States          12.0

[2716 rows x 9 columns]
```

EDA : Exploratory Data Analysis

[167]:
```
data.shape
```

[167]:
```
(2716, 9)
```

[168]:
```
data.info
```

[168]:
```
<bound method DataFrame.info of                          Company     Location HQ
Industry  # Laid Off  \
0                           D-ID        Tel Aviv              AI        22.0
```

```
1                  Zonar Systems        Seattle       Logistics       NaN
2                        Wayfair         Boston          Retail     340.0
3     Hewlett Packard Enterprise    SF Bay Area        Hardware    2500.0
4                       LiveRamp    SF Bay Area       Marketing      65.0
…                              …              …               …        …
2711                     Service    Los Angeles          Travel       NaN
2712                 HopSkipDrive   Los Angeles  Transportation       8.0
2713                  Panda Squad   SF Bay Area        Consumer       6.0
2714                Tamara Mellon   Los Angeles          Retail      20.0
2715                    EasyPost Salt Lake City       Logistics      75.0

     Percentage        Date     Stage        Country $ Raised (mm)
0           25%  2025-03-10       NaN         Israel           $48
1           NaN  2025-03-09  Acquired  United States           $50
2           NaN  2025-03-07  Post-IPO        Germany        $1,700
3            5%  2025-03-06  Post-IPO  United States        $1,400
4            5%  2025-03-05  Post-IPO  United States           $16
…          …           …         …              …            …
2711        1.0  2020-03-16      Seed  United States           5.1
2712        0.1  2020-03-13   Unknown  United States          45.0
2713       0.75  2020-03-13      Seed  United States           1.0
2714        0.4  2020-03-12  Series C  United States          90.0
2715        NaN  2020-03-11  Series A  United States          12.0

[2716 rows x 9 columns]>
```

[81]:
```
data.describe()
```

[81]:
```
          # Laid Off
count  1700.000000
mean    212.572353
std     627.861307
min       3.000000
25%      35.000000
50%      75.000000
75%     156.000000
max   10000.000000
```

[82]:
```
data.isna().sum()
```

[82]:
```
Company           0
Location HQ       0
Industry          6
# Laid Off     1016
Percentage     1019
Date              0
Stage           792
```

```
Country              0
$ Raised (mm)      210
dtype: int64
```

Data Cleaning

In the dataset, there are three important variables that require attention due to the presence of null values: "# Laid Off," "Percentage,", "$ Raised" and "Stage". Of these, "# Laid Off", "Percentage" have null values exceeding 20% of the total data points. Given that there is a good proportion of missing data in these variables, it is important to remove them from the analysis to ensure that the results are reliable.

"Raised" contains approximately 10% null values,these are not replaced by imputation of mean or median values. Similarly for "Stage"(excedding 20% null values) as well. This is because "$ Raised" and "Stage" can differ significantly between different companies and is influenced by a multitude of factors. Therefore, replacing the missing values with summary statistics could distort the data and potentially lead to erroneous conclusions.

While "#Laid Off", "Percentage" are removed due to their high null percentages, "funds_raised" is retained with its missing values intact to maintain data integrity and accuracy in subsequent analyses.

[169]:
```python
data.dropna(subset=["# Laid Off", "Percentage"], inplace=True)
data
```

[169]:
```
                         Company  Location HQ        Industry  # Laid Off  \
0                           D-ID     Tel Aviv              AI        22.0
3     Hewlett Packard Enterprise  SF Bay Area        Hardware      2500.0
4                       LiveRamp  SF Bay Area       Marketing        65.0
8                       Digimarc     Portland           Other        90.0
13               Skybox Security  SF Bay Area        Security       300.0
…                              …            …               …           …
2709                   Inspirato       Denver          Travel       130.0
2710                    Help.com       Austin         Support        16.0
2712                 HopSkipDrive  Los Angeles  Transportation         8.0
2713                 Panda Squad  SF Bay Area        Consumer         6.0
2714               Tamara Mellon  Los Angeles          Retail        20.0

     Percentage        Date            Stage        Country $ Raised (mm)
0           25%  2025-03-10              NaN         Israel           $48
3            5%  2025-03-06         Post-IPO  United States        $1,400
4            5%  2025-03-05         Post-IPO  United States           $16
8           40%  2025-02-27         Post-IPO  United States          $105
13         100%  2025-02-24  Private Equity  United States          $335
…             …           …                …              …             …
2709       0.22  2020-03-16         Series C  United States          79.0
2710        1.0  2020-03-16             Seed  United States           6.0
2712        0.1  2020-03-13          Unknown  United States          45.0
2713       0.75  2020-03-13             Seed  United States           1.0
```

```
2714        0.4  2020-03-12       Series C  United States          90.0

[1238 rows x 9 columns]
```

[170]:
```
data.isna().sum()
```

[170]:
```
Company            0
Location HQ        0
Industry           3
# Laid Off         0
Percentage         0
Date               0
Stage            333
Country            0
$ Raised (mm)    131
dtype: int64
```

[171]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 1238 entries, 0 to 2714
Data columns (total 9 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Company        1238 non-null   object
 1   Location HQ    1238 non-null   object
 2   Industry       1235 non-null   object
 3   # Laid Off     1238 non-null   float64
 4   Percentage     1238 non-null   object
 5   Date           1238 non-null   object
 6   Stage          905 non-null    object
 7   Country        1238 non-null   object
 8   $ Raised (mm)  1107 non-null   object
dtypes: float64(1), object(8)
memory usage: 96.7+ KB
```

[172]:
```
data.Company.nunique()
```

[172]:
```
1142
```

[173]:
```
data["Location HQ"].nunique()
```

[173]:
```
142
```

[174]:
```
data["Location HQ"].unique()
```

[174]:
```
array(['Tel Aviv', 'SF Bay Area', 'Portland', 'Denver', 'New York City',
       'Manchester', 'Lisbon', 'Seattle', 'Vancouver', 'Munich',
```

```
       'Rochester', 'Singapore', 'London', 'Chicago', 'Bengaluru',
       'San Diego', 'Stockholm', 'Dublin', 'Los Angeles', 'Boston',
       "Ra'anana", 'Washington D.C.', 'Durham', 'Toronto', 'Norwalk',
       'Dallas', 'Frankfurt', 'Austin', 'Chennai', 'Dover', 'Montevideo',
       'New Delhi', 'Orlando', 'Prague', 'Ghent', 'Copenhagen', 'Lehi',
       'Pittsburgh', 'Nashik', 'Oslo', 'Brno', 'Boulder', 'Saskatoon',
       'Miami', 'Corvallis', 'Berlin', 'Milwaukee', 'Haifa', 'Mumbai',
       'Detroit', 'Noida', 'Calgary', 'Amsterdam', 'Paris', 'Kansas City',
       'Jacksonville', 'Montreal', 'Raleigh', 'Salt Lake City',
       'Belo Horizonte', 'Tallinn', 'Kolkata', 'Gurugram', 'Sandnes',
       'Atlanta', 'Edinburgh', 'Sao Paulo', 'Auckland', 'Nairobi',
       'Phoenix', 'Tampa Bay', 'Lagos', 'St. Louis', 'Sydney', 'Seoul',
       'Baltimore', 'Melbourne', 'Charlottesville', 'Las Vegas',
       'Santiago', 'Ottawa', 'Brisbane', 'Riyadh', 'Wrocław', 'Cleveland',
       'Philadelphia', 'Nashua', 'Chester', 'Linz', 'Madison',
       'Wellington', 'Pune', 'Albany', 'Charleston', 'Jakarta',
       'Columbus', 'Karlsruhe', 'Barcelona', 'Walldorf', 'Reno', 'Kiel',
       'Oxford', 'Wilmington', 'Tokyo', 'Beijing', 'Cincinnati',
       'Hamburg', 'Utrecht', 'Waterloo', 'Buenos Aires', 'Luxembourg',
       'Nebraska City', 'Stamford', 'San Luis Obispo', 'Jerusalem',
       'Bangkok', 'Indianapolis', 'Ferdericton', 'Florianópolis', 'Dakar',
       'Hong Kong', 'Curitiba', 'Helsinki', 'Bend', 'Brasilia', 'Dubai',
       'Non-U.S.', 'Nashville', 'Sofia', 'Santa Fe', 'Spokane',
       'Ahmedabad', 'Joinville', 'Zurich', 'Missoula', 'Minneapolis',
       'Blumenau', 'Guadalajara', 'Ann Arbor', 'Kuala Lumpur', 'Yangon',
       'Sacramento'], dtype=object)
```

[175]:
```
data.Stage.nunique()
```

[175]:
```
16
```

[176]:
```
data.Stage.unique()
```

[176]:
```
array([nan, 'Post-IPO', 'Private Equity', 'Series E', 'Series D',
       'Unknown', 'Series B', 'Subsidiary', 'Series F', 'Acquired',
       'Series C', 'Series A', 'Series G', 'Series H', 'Seed', 'Series I',
       'Series J'], dtype=object)
```

[177]:
```
data.Country.nunique()
```

[177]:
```
44
```

[178]:
```
data.Country.unique()
```

[178]:
```
array(['Israel', 'United States', 'United Kingdom', 'Portugal', 'Canada',
       'Germany', 'Singapore', 'India', 'Sweden', 'Ireland', 'Uruguay',
       'Czech Republic', 'Belgium', 'Denmark', 'Norway', 'Netherlands',
       'France', 'Brazil', 'Estonia', 'New Zealand', 'Kenya', 'Nigeria',
```

```
       'Australia', 'South Korea', 'Chile', 'Saudi Arabia', 'Poland',
       'Austria', 'Indonesia', 'Spain', 'China', 'Argentina',
       'Luxembourg', 'Thailand', 'Senegal', 'Hong Kong', 'Finland',
       'United Arab Emirates', 'Seychelles', 'Bulgaria', 'Switzerland',
       'Mexico', 'Malaysia', 'Myanmar'], dtype=object)
```

Extracting the top 10 companies that laid off for further analysis

[179]:
```
company_layoffs = data.groupby('Company')['# Laid Off'].sum().reset_index()
top_companies = company_layoffs.nlargest(10, '# Laid Off')
print(top_companies)
```

```
          Company  # Laid Off
615     Microsoft     10000.0
314      Ericsson      8500.0
40         Amazon      8000.0
348         Flink      8000.0
1025         Uber      7525.0
261          Dell      6650.0
122   Booking.com      4375.0
733       Philips      4000.0
97     Better.com      3900.0
1021      Twitter      3700.0
```

Visualizing the Data

[180]:
```
import plotly.express as px
import pandas as pd

# Aggregate total layoffs by country
layoffs_by_country = data.groupby('Country')['# Laid Off'].sum().reset_index()

#Processing
layoffs_by_country['Layoffs_Label'] = layoffs_by_country['# Laid Off'].apply(
    lambda x: f"{int(x/1000)}K" if x >= 1000 else f"{int(x)}"
)

#Define plot
fig = px.scatter_geo(
    layoffs_by_country,
    locations="Country",
    locationmode="country names",
    size="# Laid Off",  # Bubble size represents layoffs
    hover_name="Country",
    text=layoffs_by_country["Layoffs_Label"],  # Display layoffs in "K" format
    color="# Laid Off",
    color_continuous_scale="Reds",  # Color gradient for layoffs
    title="Global Tech Layoffs Impact by Country (2020-2025)",
    projection="natural earth"
```

```
)

fig.update_traces(
    textposition="top center",
    marker=dict(line=dict(color="black", width=1))
)

fig.update_layout(
    geo=dict(
        showcoastlines=True,
        showland=True,
        landcolor="lightgray",
        countrycolor="black"
    )
)

# Show the interactive map
fig.show()
```

[181]:
```
data_with_date_index = data.set_index('Date')

# Filter data for each year
data_2025 = data_with_date_index.loc[(data_with_date_index.index >␣
↪'2025-01-01')&(data_with_date_index.index < '2026-01-01')]
data_2024 = data_with_date_index.loc[(data_with_date_index.index >␣
↪'2024-01-01')&(data_with_date_index.index < '2025-01-01')]
data_2023 = data_with_date_index.loc[(data_with_date_index.index >␣
↪'2023-01-01')&(data_with_date_index.index < '2024-01-01')]
data_2022 = data_with_date_index.loc[(data_with_date_index.index >␣
↪'2022-01-01')&(data_with_date_index.index < '2023-01-01')]
data_2021 = data_with_date_index.loc[(data_with_date_index.index >␣
↪'2021-01-01')&(data_with_date_index.index < '2022-01-01')]
data_2020 = data_with_date_index.loc[(data_with_date_index.index >␣
↪'2020-01-01')&(data_with_date_index.index < '2021-01-01')]
```

[182]:
```
# Combine data from all years
combined_data = pd.concat([data_2020, data_2021, data_2022, data_2023,␣
↪data_2024, data_2025])

combined_data.index = pd.to_datetime(combined_data.index)
combined_data['Year'] = combined_data.index.year

# Aggregate total layoffs
annual_layoffs = combined_data.groupby('Year')['# Laid Off'].sum().reset_index()

#Plot it
layoffs_plot = px.line(
```

```python
    annual_layoffs,
    x='Year',
    y='# Laid Off',
    title='Trends of Layoffs Over Five Years',
    labels={'Year': 'Year', '# Laid Off': 'Total Layoffs'},
    color_discrete_sequence=px.colors.sequential.Plasma,
)

layoffs_plot.update_traces(mode='lines+markers', marker=dict(size=12))

layoffs_plot.update_layout(
    autosize=False,
    width=800,
    height=500,
    xaxis=dict(dtick=1),
    showlegend=False,
)

layoffs_plot.show()
```

Global Layoffs Trends Since the start of COVID-19 (Yearly)

This line chart visualizes the fluctuations in global layoffs from 2020 to 2025, highlighting key shifts in employment patterns. Markers on the line indicate yearly totals, capturing the broader economic and industry-specific impacts on workforce reductions.

Observations:

2020: Initial Shock In 2020, A significant rise in layoffs (70K+) occurred as businesses faced the sudden impact of COVID-19, lockdowns, and economic downturns.

2021: Temporary Recovery Layoffs dropped sharply (6.5K), reflecting early signs of economic stabilization and adaptation to new business environments.

2022: Tech Sector Disruptions A resurgence in layoffs (139K) suggests economic uncertainty, hiring corrections, and shifts in workforce demands, particularly in tech.

2023: Peak Layoff Year The highest layoffs (181K) were recorded, driven by corporate downsizing and cost-cutting measures after a hiring boom.

2024-2025: Market Correction Layoff figures decline, indicating economic adjustments, corporate restructuring, and long-term stabilization.

Significance:

This chart provides critical insights into the ever-changing employment landscape during the pandemic. Policymakers and businesses can use it as a valuable reference point while navigating the dynamic effects of COVID-19 on the global workforce and the implications of mass hiring by tech companies.

Layoff Trends Over Time: A Yearly Breakdown (2020-2024)

To understand the global layoff landscape from 2020 to 2024, we analyze the most significant workforce reductions each year. Starting with 2024 (due to limited data for 2025) and working backward to 2020, this approach highlights key employment shifts and industry trends. By examining how layoffs have evolved, we can uncover patterns in economic recovery, corporate restructuring, and sector-specific downturns, offering a clearer picture of workforce dynamics over time.

[191]:
```
df_2024_most_layoffs = data_2024.sort_values(by='# Laid Off', ascending=False)
df_2024_most_layoffs.head()
```

[191]:
```
              Company  Location HQ  Industry  # Laid Off Percentage     Stage  \
Date
2024-01-03      Xerox      Norwalk  Hardware      3000.0        15%  Post-IPO
2024-02-16   Farfetch       London    Retail      2000.0        25%       NaN
2024-01-08      Unity  SF Bay Area     Other      1800.0       0.25  Post-IPO
2024-07-10     Intuit  SF Bay Area   Finance      1800.0        10%  Post-IPO
2024-09-23  Northvolt    Stockholm    Energy      1600.0        20%       NaN

                   Country $ Raised (mm)
Date
2024-01-03   United States       $27,200
2024-02-16  United Kingdom        $1,700
2024-01-08   United States        1300.0
2024-07-10   United States           $18
2024-09-23          Sweden       $13,800
```

[192]:
```
df_2023_most_layoffs = data_2023.sort_values(by='# Laid Off', ascending=False)
df_2023_most_layoffs.head()
```

[192]:
```
              Company Location HQ  Industry  # Laid Off Percentage     Stage  \
Date
2023-01-18  Microsoft     Seattle     Other     10000.0         5%  Post-IPO
2023-02-24   Ericsson   Stockholm     Other      8500.0         8%       NaN
2023-04-24      Flink      Berlin      Food      8000.0        40%       NaN
2023-01-04     Amazon     Seattle    Retail      8000.0         2%  Post-IPO
2023-02-06       Dell      Austin  Hardware      6650.0         5%  Post-IPO

                  Country $ Raised (mm)
Date
2023-01-18  United States            $1
2023-02-24         Sweden          $663
2023-04-24        Germany        $1,000
2023-01-04  United States          $108
2023-02-06  United States           NaN
```

[185]:
```
df_2022_most_layoffs = data_2022.sort_values(by='# Laid Off', ascending=False)
df_2022_most_layoffs.head()
```

[185]:

```
               Company     Location HQ        Industry  # Laid Off Percentage  \
Date
2022-10-24     Philips       Amsterdam      Healthcare      4000.0          5%
2022-11-04     Twitter     SF Bay Area        Consumer      3700.0         50%
2022-03-08  Better.com   New York City     Real Estate      3000.0         33%
2022-02-08     Peloton   New York City         Fitness      2800.0         20%
2022-05-10     Carvana         Phoenix  Transportation      2500.0         12%

               Stage        Country $ Raised (mm)
Date
2022-10-24       NaN    Netherlands           NaN
2022-11-04  Post-IPO  United States       $12,900
2022-03-08   Unknown  United States          $905
2022-02-08  Post-IPO  United States        $1,900
2022-05-10  Post-IPO  United States        $1,600
```

[186]:

```python
df_2021_most_layoffs = data_2021.sort_values(by='# Laid Off', ascending=False)
df_2021_most_layoffs.head()
```

[186]:

```
               Company     Location HQ        Industry  # Laid Off Percentage  \
Date
2021-06-01     Katerra     SF Bay Area    Construction      2434.0         1.0
2021-11-02      Zillow         Seattle     Real Estate      2000.0        0.25
2021-12-01  Better.com   New York City     Real Estate       900.0        0.09
2021-01-13     Dropbox     SF Bay Area           Other       315.0        0.15
2021-02-22      Bounce       Bengaluru  Transportation       200.0         0.4

               Stage        Country $ Raised (mm)
Date
2021-06-01   Unknown  United States        1600.0
2021-11-02  Post-IPO  United States          97.0
2021-12-01   Unknown  United States         905.0
2021-01-13  Post-IPO  United States        1700.0
2021-02-22  Series D          India         214.2
```

[187]:

```python
df_2020_most_layoffs = data_2020.sort_values(by='# Laid Off', ascending=False)
df_2020_most_layoffs.head()
```

[187]:

```
                Company  Location HQ        Industry  # Laid Off Percentage  \
Date
2020-07-30  Booking.com    Amsterdam          Travel      4375.0        0.25
2020-05-06         Uber  SF Bay Area  Transportation      3700.0        0.14
2020-05-18         Uber  SF Bay Area  Transportation      3000.0        0.13
2020-04-13      Groupon      Chicago          Retail      2800.0        0.44
2020-05-05       Airbnb  SF Bay Area          Travel      1900.0        0.25

                     Stage        Country $ Raised (mm)
```

```
Date
2020-07-30        Acquired    Netherlands          NaN
2020-05-06        Post-IPO  United States      24700.0
2020-05-18        Post-IPO  United States      24700.0
2020-04-13        Post-IPO  United States       1400.0
2020-05-05  Private Equity  United States       5400.0
```

Global Layoff Impact: Countries Facing the Highest Job Cuts

As part of our in-depth analysis of worldwide layoffs since the COVID-19 pandemic, a notable trend has emerged: the majority of major workforce reductions each year have been driven by companies headquartered in the United States. This highlights the substantial effect of economic shifts on the U.S. job market.

To gain a broader perspective, we are extending our focus to examine the nations most impacted by layoffs outside the U.S. By visualizing job cuts across different regions, this analysis aims to offer a more comprehensive understanding of the global employment landscape, revealing which countries have experienced the most significant workforce reductions during this turbulent period.

[193]:
```python
import pandas as pd
import plotly.express as px

#Create Year
final_df['Year'] = pd.to_datetime(final_df['Date']).dt.year  # Extract year␣
 ↪from Date

# Aggregate layoffs
layoffs_by_year_country = final_df.groupby(['Year', 'Country'])['# Laid Off'].
 ↪sum().reset_index()

fig = px.treemap(
    layoffs_by_year_country,
    path=['Year', 'Country'],  # Year → Country hierarchy
    values='# Laid Off',
    title="Layoffs Breakdown by Year and Country (2020-2025)",
    color='# Laid Off',
    color_continuous_scale='Viridis'
)

fig.show()
```

Assessing Regional Impact: Layoffs Across the United States

Building on insights from our broader analysis, several countries—including the United States, Sweden, Germany, and the Netherlands—have experienced significant layoffs from the onset of the COVID-19 pandemic through 2023.

To delve deeper into the U.S. labor market, our focus now shifts to identifying the top five locations within the United States that have faced the highest number of layoffs. This analysis will highlight regional disparities and shed light on the areas most affected by workforce reductions.

The United States of America:

[194]:
```python
import plotly.express as px
import pandas as pd

#Clean
top_5_locations['Percentage'] = top_5_locations['Percentage'].astype(str)

top_5_locations['Percentage'] = top_5_locations['Percentage'].str.
    ↪extract(r'(\d+\.?\d*)')[0]

top_5_locations['Percentage'] = pd.to_numeric(top_5_locations['Percentage'],␣
    ↪errors='coerce')

top_5_locations = top_5_locations.dropna(subset=['Percentage'])

pie_chart = px.pie(
    top_5_locations,
    names='Location HQ',
    values='Percentage',
    color_discrete_sequence=px.colors.sequential.Sunset,
    title='Top 5 Locations in the United States with the Highest Layoffs',
    hole=0.3,
)

# Add hover data
pie_chart.update_traces(
    hoverinfo='label+percent+value',
    textinfo='percent',
    marker=dict(line=dict(color='#000000', width=2)),
)


pie_chart.show()
```

Analyzing Industry-Wise Layoffs in the San Francisco Bay Area

The San Francisco Bay Area has emerged as one of the most affected regions in terms of layoffs, driven by a mix of economic, industry-specific, and structural challenges.

Key Factors Driving Layoffs:

1. Financial Sector Sensitivity:
With a strong presence of financial institutions, tech firms, and startups, workforce adjustments in response to market fluctuations are common, influencing layoff patterns significantly.

2. High Cost of Living:
The region's steep living expenses, particularly in housing, add pressure on businesses, making cost-cutting measures, including layoffs, a frequent response to economic downturns.

3. Pandemic Disruptions in Service-Based Industries:
Sectors relying on in-person interactions—such as hospitality, transportation, and retail—faced heavy job cuts due to COVID-19-related restrictions and shifts in consumer behavior.

4. Startup Volatility:
While the Bay Area thrives on innovation, startups are highly vulnerable to economic uncertainty. Periods of aggressive expansion often lead to over-hiring, followed by mass layoffs during financial contractions.

[195]:
```
# Filter data for SF Bay Area
sf_bay_area_data = data[data['Location HQ'] == 'SF Bay Area']

top_7_industries = (
    sf_bay_area_data.groupby('Industry')
    .sum()
    .sort_values(by='Percentage', ascending=False)
    .head(7)
    .reset_index()
)
```

[196]:
```
import plotly.express as px

top_7_industries['Percentage'] = pd.to_numeric(
    top_7_industries['Percentage'].str.extract(r'(\d+\.?\d*)')[0],␣
↪errors='coerce'
)

fig = px.scatter(
    top_7_industries,
    x='Industry',
    y='Percentage',
    color='Percentage',
    color_continuous_scale=px.colors.sequential.Viridis,
    size='Percentage',
    title='Top 7 Industries in Tech in the SF Bay Area with Highest Layoff␣
↪Percentages',
    labels={'Industry': 'Industry', 'Percentage': 'Layoff Percentage'},
)

fig.add_traces(
    px.line(
        top_7_industries,
        x='Industry',
        y='Percentage',
        line_shape='vh',  # Vertical-Horizontal for lollipop effect
    ).data
)
```

```python
fig.update_traces(marker=dict(line=dict(width=2, color='black')))  # Add marker␣
↪borders
fig.update_layout(
    autosize=False,
    width=800,
    height=500,
    xaxis_tickangle=-45,  # Rotate x-axis labels for readability
    plot_bgcolor='white',
    paper_bgcolor='white',
)

# Show the plot
fig.show()
```

[197]:
```python
# Filter data for SF Bay Area and Finance industry
sf_bay_area_data = data[data['Location HQ'] == 'SF Bay Area']
finance_industry_data = data[data['Industry'] == 'Finance']

selected_finance_data = finance_industry_data[['Company', '# Laid Off']].
↪reset_index(drop=True)

top_10_companies = (
    finance_industry_data.groupby('Company')
    .sum()
    .sort_values(by='# Laid Off', ascending=False)
    .head(10)
    .reset_index()
)
```

[135]:
```python
import plotly.express as px

# Ensure data is loaded
if 'final_df' in locals():
    # Extract top 10 companies by total layoffs
    top_10_companies = final_df.groupby('Company')['# Laid Off'].sum().
↪reset_index()
    top_10_companies = top_10_companies.sort_values(by='# Laid Off',␣
↪ascending=False).head(10)

    # Create a horizontal bar chart
    fig = px.bar(
        top_10_companies,
        x='# Laid Off',
        y='Company',
        orientation='h',  # Horizontal bars
        title='Top 10 Companies in the FinTech Industry with the Highest␣
↪Layoffs',
```

```
        labels={'Company': 'Company', '# Laid Off': 'Total Laid Off'},
        color='# Laid Off',  # Color based on layoffs
        color_continuous_scale=px.colors.sequential.Viridis,  # Use a different␣
↪color scheme
        text='# Laid Off',  # Show the number of layoffs on bars
    )

    # Update layout for better readability
    fig.update_layout(
        autosize=False,
        width=800,
        height=500,
        xaxis_title_font=dict(size=14, color='black'),
        yaxis_title_font=dict(size=14, color='black'),
        font=dict(size=12, color='black'),
        plot_bgcolor='white',
        paper_bgcolor='white',
        yaxis=dict(autorange="reversed"),  # Reverse y-axis for top-down ranking
    )

    # Show the plot
    fig.show()
else:
    print("Error: DataFrame 'final_df' is not loaded. Please check your dataset.
↪")
```