# Week 2

## Practical Data Transformation

Machine Learning Graz

# Agenda

1. Data Reading & Data Cleaning

   Break

2. Exploratory Data Analysis

# Grading

70% - Final Project

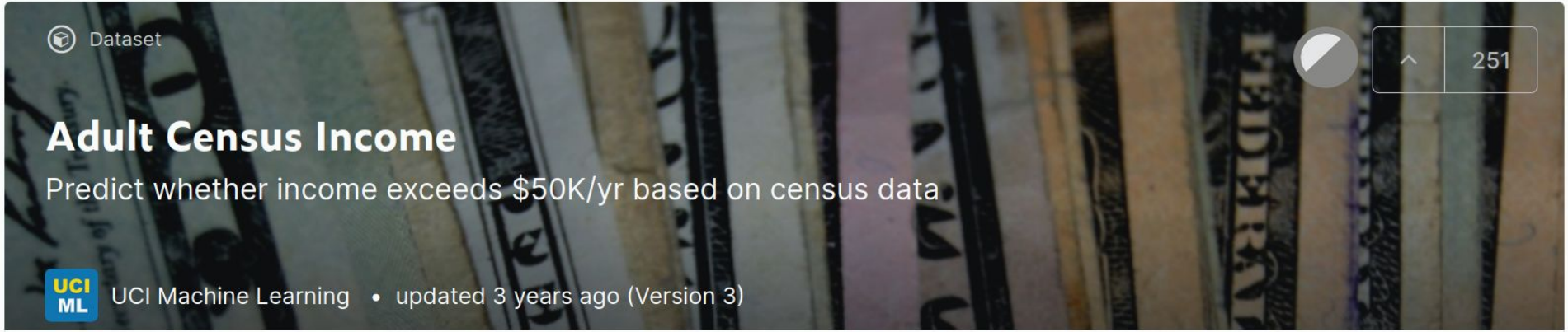
20% - Assignments

10% - Live Quizzes

+10 Points for every invited person.

# What is Google Collab?

# What data are we gonna work today?




Data Tasks Kernels (244) Discussion (8) Activity Metadata Download (4 MB) New Notebook

Usability 7.1 License CC0: Public Domain Tags reference, social sciences, mathematics, utility, demographics and 1 more


.DOC
.JPG
.PNG
.PSD
.EPS
.CDR
.TXT
.GIF
.PPT
.MP3
.WAV
.AI
.MOV
.EXE
.DMG
.RAR
.ZIP
.PDF

# Common Data Science Formats

- Comma-separated values (csv)
- XLSX
- ZIP
- Plain Text (txt)
- JSON
- HTML
- SQL
- Images
- Hierarchical Data Format
- PDF
- DOCX
- MP3
- MP4

# Spreadsheet Format

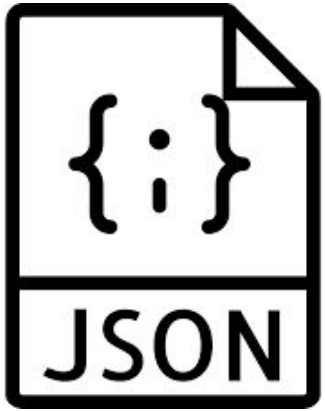
In spreadsheet format, data is stored in cells. Each cell is organized in rows and columns. A column in the spreadsheet file can have different types.

| Street Name | Amount of Rooms | Area | Type | Price |
|---|---|---|---|---|
| Glacisstrasse | 5 | 220 | Luxury | 2500000 |
| Inffeldgasse | 3 | 120 | Standart | 450000 |
| Kasernstrasse | 2 | 45 | Econom | 100000 |
| Kreuzgasse | 6 | 250 | Luxury | 3000000 |

# Common File Formats for Spreadsheets

- CSV
- XLSX
- JSON
- HTML





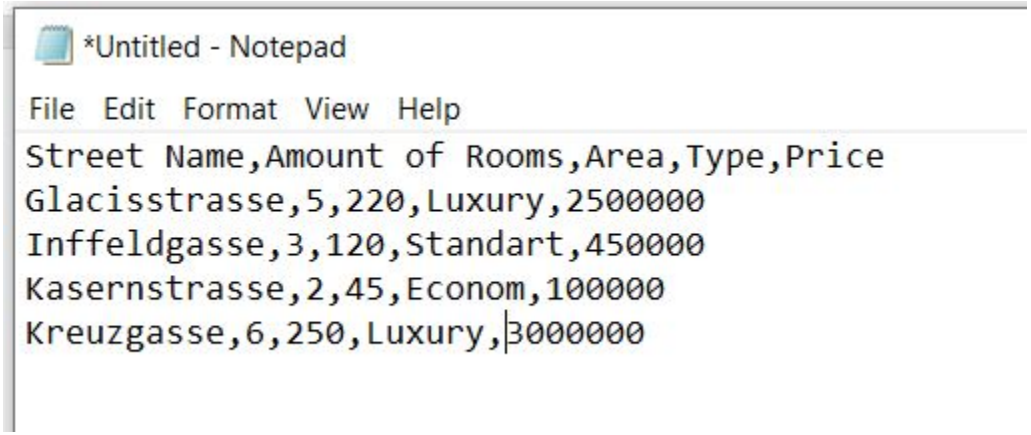

# How to read all this data formats?

# How to read all this data formats?

# How to read all this data formats?

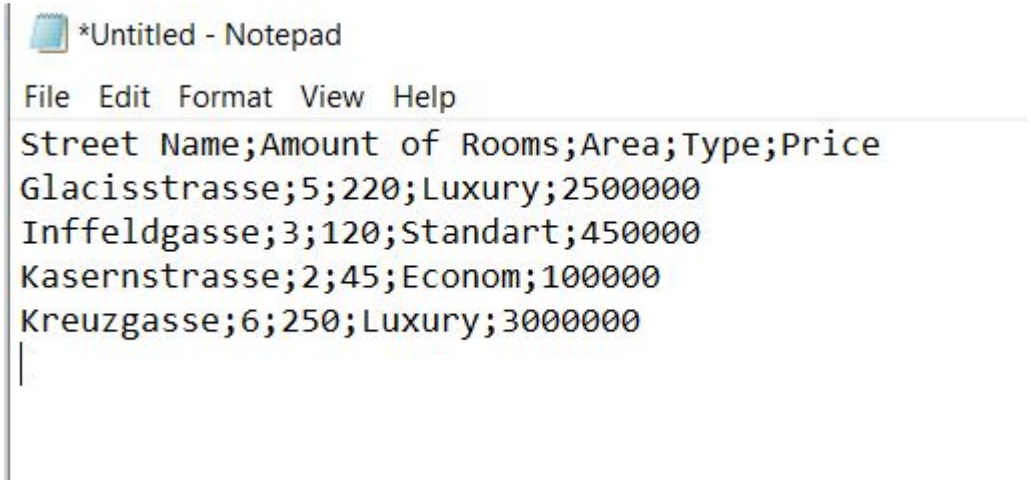
# CSV file format

Each line in CSV file represents an observation or commonly called a record.
Each record may contain one or more fields which are separated by a comma.



# CSV file format

Sometimes you may come across files where fields are not separated by using a comma but with other delimiter. For example: ‘ ; ’, ‘ \t ’, ‘ # ’ etc.



# How to read CSV file?

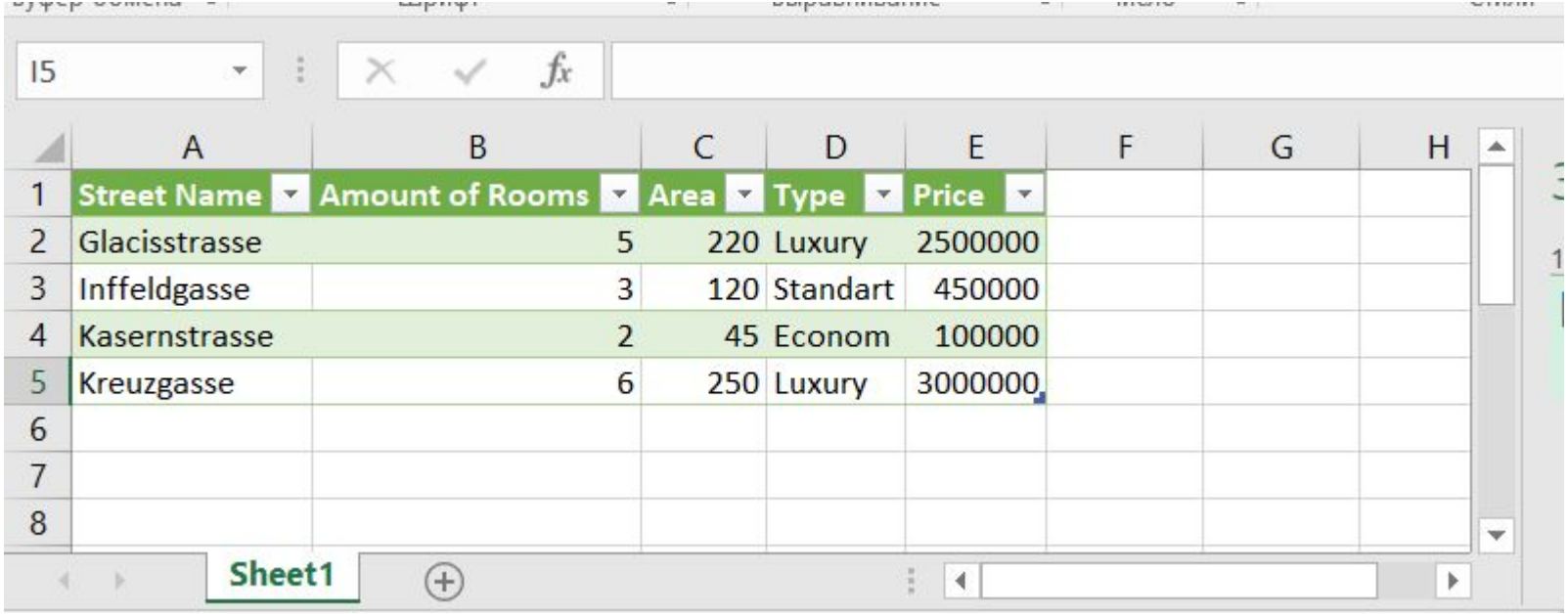
```
In [1]: import pandas as pd

In [2]: df = pd.read_csv('example.csv', sep=',')

In [3]: df
```

Out[3]:

| | Street Name | Amount of Rooms | Area | Type | Price |
|---|---|---|---|---|---|
| **0** | Glacisstrasse | 5 | 220 | Luxury | 2500000 |
| **1** | Inffeldgasse | 3 | 120 | Standart | 450000 |
| **2** | Kasernstrasse | 2 | 45 | Econom | 100000 |
| **3** | Kreuzgasse | 6 | 250 | Luxury | 3000000 |

# XLSX file format

XLSX is a Microsoft Excel Open XML file format. It is an XML-based file format created by Microsoft Excel.



| Street Name | Amount of Rooms | Area | Type | Price |
|---|---|---|---|---|
| Glacisstrasse | 5 | 220 | Luxury | 2500000 |
| Inffeldgasse | 3 | 120 | Standart | 450000 |
| Kasernstrasse | 2 | 45 | Econom | 100000 |
| Kreuzgasse | 6 | 250 | Luxury | 3000000 |

Sheet1

# How to read XLSX?

```
In [1]: import pandas as pd
```

```
In [2]: df = pd.read_excel('example.xlsx', sheet_name='Sheet1')
```

```
In [3]: df
```

Out[3]:

| | Street Name | Amount of Rooms | Area | Type | Price |
|---|---|---|---|---|---|
| **0** | Glacisstrasse | 5 | 220 | Luxury | 2500000 |
| **1** | Inffeldgasse | 3 | 120 | Standart | 450000 |
| **2** | Kasernstrasse | 2 | 45 | Econom | 100000 |
| **3** | Kreuzgasse | 6 | 250 | Luxury | 3000000 |

# JSON file format

JavaScript Object Notation(JSON) is a text-based open standard designed for exchanging the data over web.

*example - Notepad

File Edit Format View Help


```
{"Street Name":{"0":"Glacisstrasse","1":"Inffeldgasse","2":"Kasernstrasse","3":"Kreuzgasse"},
"Amount of Rooms":{"0":5,"1":3,"2":2,"3":6},"Area":{"0":220,"1":120,"2":45,"3":250},
"Type":{"0":"Luxury","1":"Standart","2":"Econom","3":"Luxury"},
"Price":{"0":2500000,"1":450000,"2":100000,"3":3000000}}
```

# How to read JSON?

```
In [1]: import pandas as pd

In [7]: df = pd.read_json('example.json')

In [8]: df
```

Out[8]:

| | Street Name | Amount of Rooms | Area | Type | Price |
|---|---|---|---|---|---|
| **0** | Glacisstrasse | 5 | 220 | Luxury | 2500000 |
| **1** | Inffeldgasse | 3 | 120 | Standart | 450000 |
| **2** | Kasernstrasse | 2 | 45 | Econom | 100000 |
| **3** | Kreuzgasse | 6 | 250 | Luxury | 3000000 |

# HTML file format

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser.


```
'<table border="1" class="dataframe">\n  <thead>\n    <tr style="text-align: right;">\n      <th>Street Name</th>\n
<th>Amount of Rooms</th>\n      <th>Area</th>\n      <th>Type</th>\n      <th>Price</th>\n    </tr>\n  </thead>\n  <t
body>\n    <tr>\n      <td>Glacisstrasse</td>\n      <td>5</td>\n      <td>220</td>\n      <td>Luxury</td>\n      <td
>2500000</td>\n    </tr>\n    <tr>\n      <td>Inffeldgasse</td>\n      <td>3</td>\n      <td>120</td>\n      <td>Stan
dart</td>\n      <td>450000</td>\n    </tr>\n    <tr>\n      <td>Kasernstrasse</td>\n      <td>2</td>\n      <td>45</
td>\n      <td>Econom</td>\n      <td>100000</td>\n    </tr>\n    <tr>\n      <td>Kreuzgasse</td>\n      <td>6</td>\n
<td>250</td>\n      <td>Luxury</td>\n      <td>3000000</td>\n    </tr>\n  </tbody>\n</table>'
```

# How to read HTML table?

```
In [1]: import pandas as pd
```

```
In [11]: df = pd.read_html('example.html')[0]
```

```
In [12]: df
```

Out[12]:

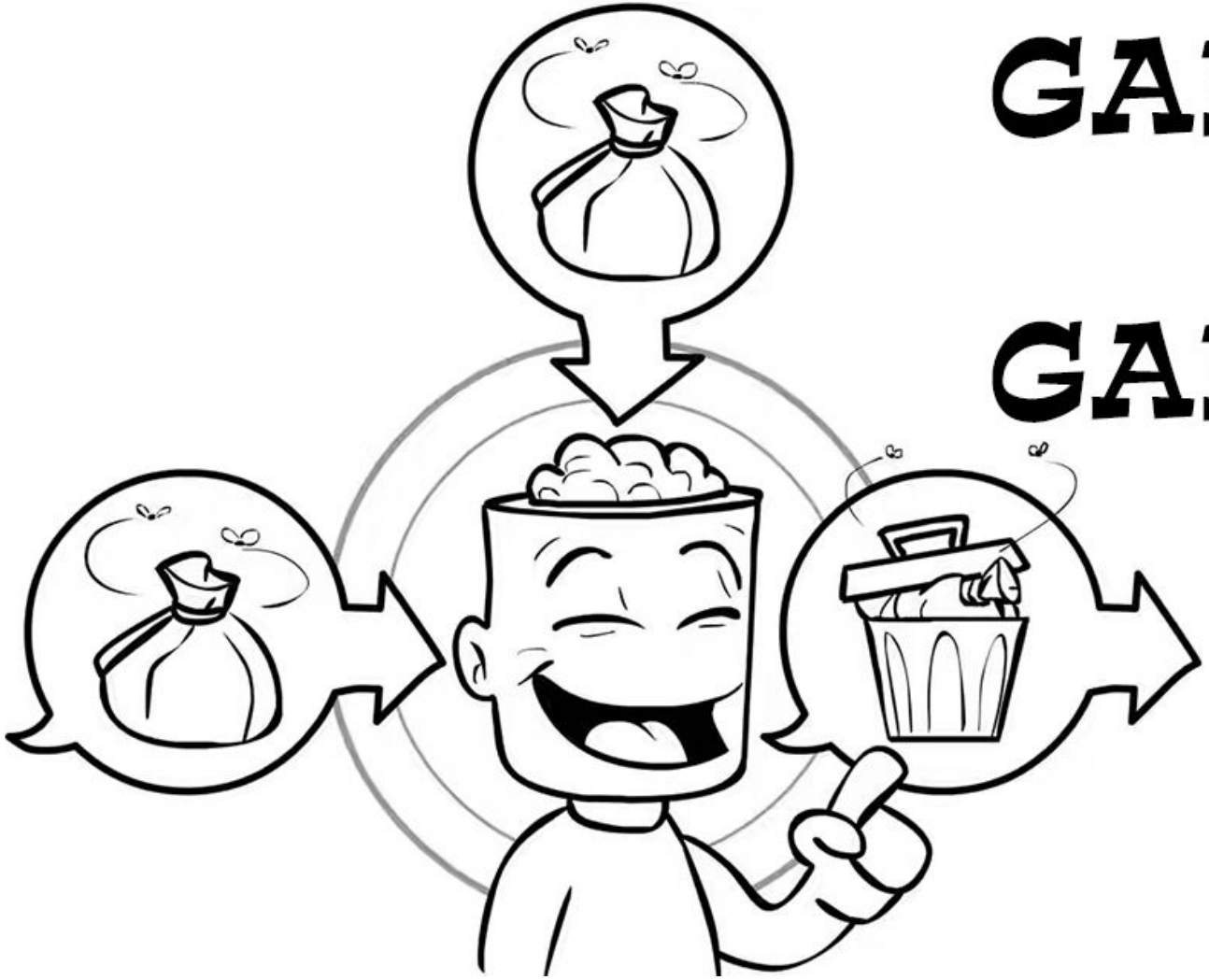
| | Street Name | Amount of Rooms | Area | Type | Price |
|---|---|---|---|---|---|
| **0** | Glacisstrasse | 5 | 220 | Luxury | 2500000 |
| **1** | Inffeldgasse | 3 | 120 | Standart | 450000 |
| **2** | Kasernstrasse | 2 | 45 | Econom | 100000 |
| **3** | Kreuzgasse | 6 | 250 | Luxury | 3000000 |

# Pandas IO tools

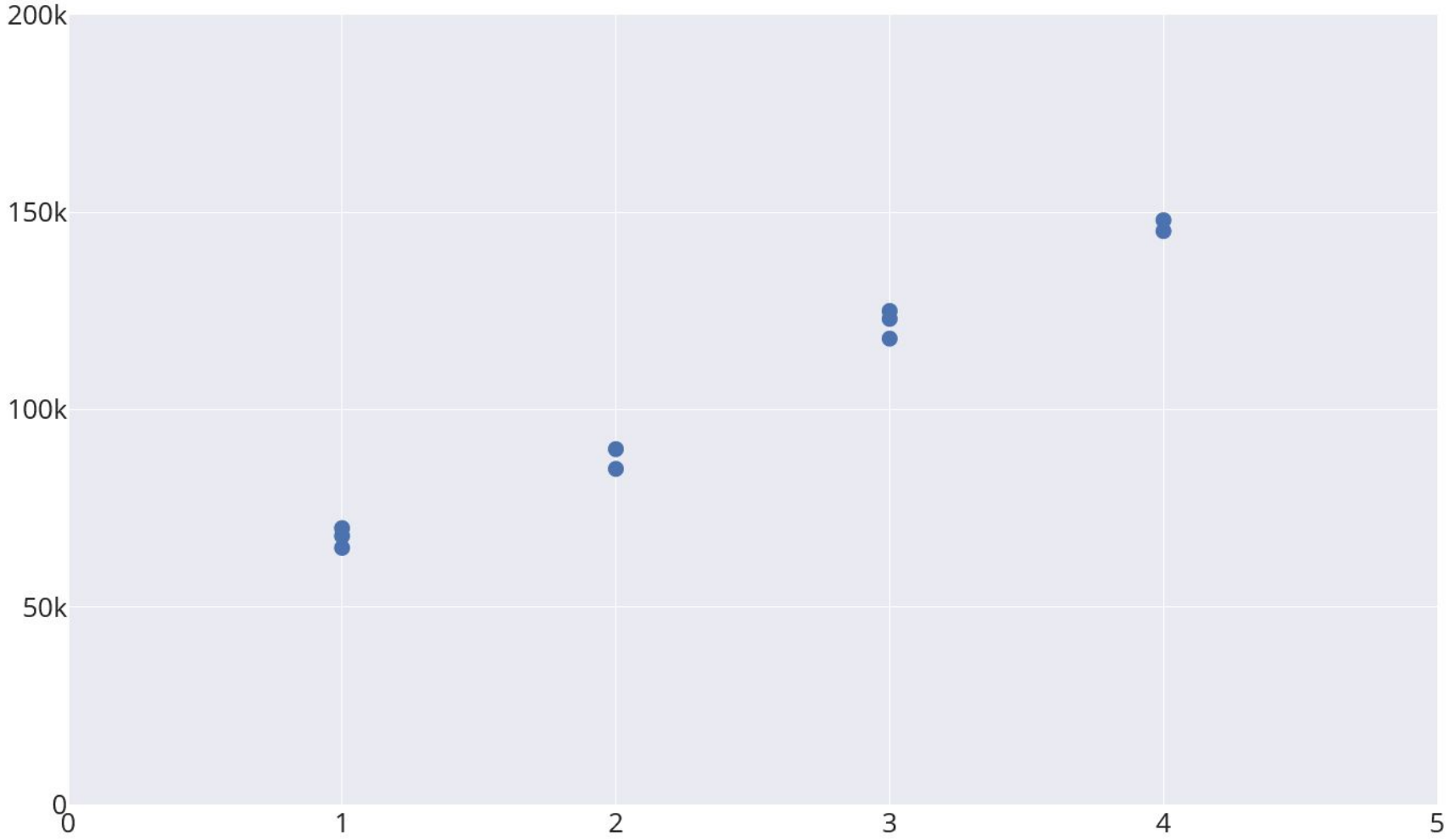
| Format Type | Data Description | Reader | Writer |
|---|---|---|---|
| text | CSV | read_csv | to_csv |
| text | JSON | read_json | to_json |
| text | HTML | read_html | to_html |
| text | Local clipboard | read_clipboard | to_clipboard |
| binary | MS Excel | read_excel | to_excel |
| binary | HDF5 Format | read_hdf | to_hdf |
| binary | Feather Format | read_feather | to_feather |
| binary | Msgpack | read_msgpack | to_msgpack |
| binary | Stata | read_stata | to_stata |
| binary | SAS | read_sas | |
| binary | Python Pickle Format | read_pickle | to_pickle |
| SQL | SQL | read_sql | to_sql |
| SQL | Google Big Query | read_gbq | to_gbq |

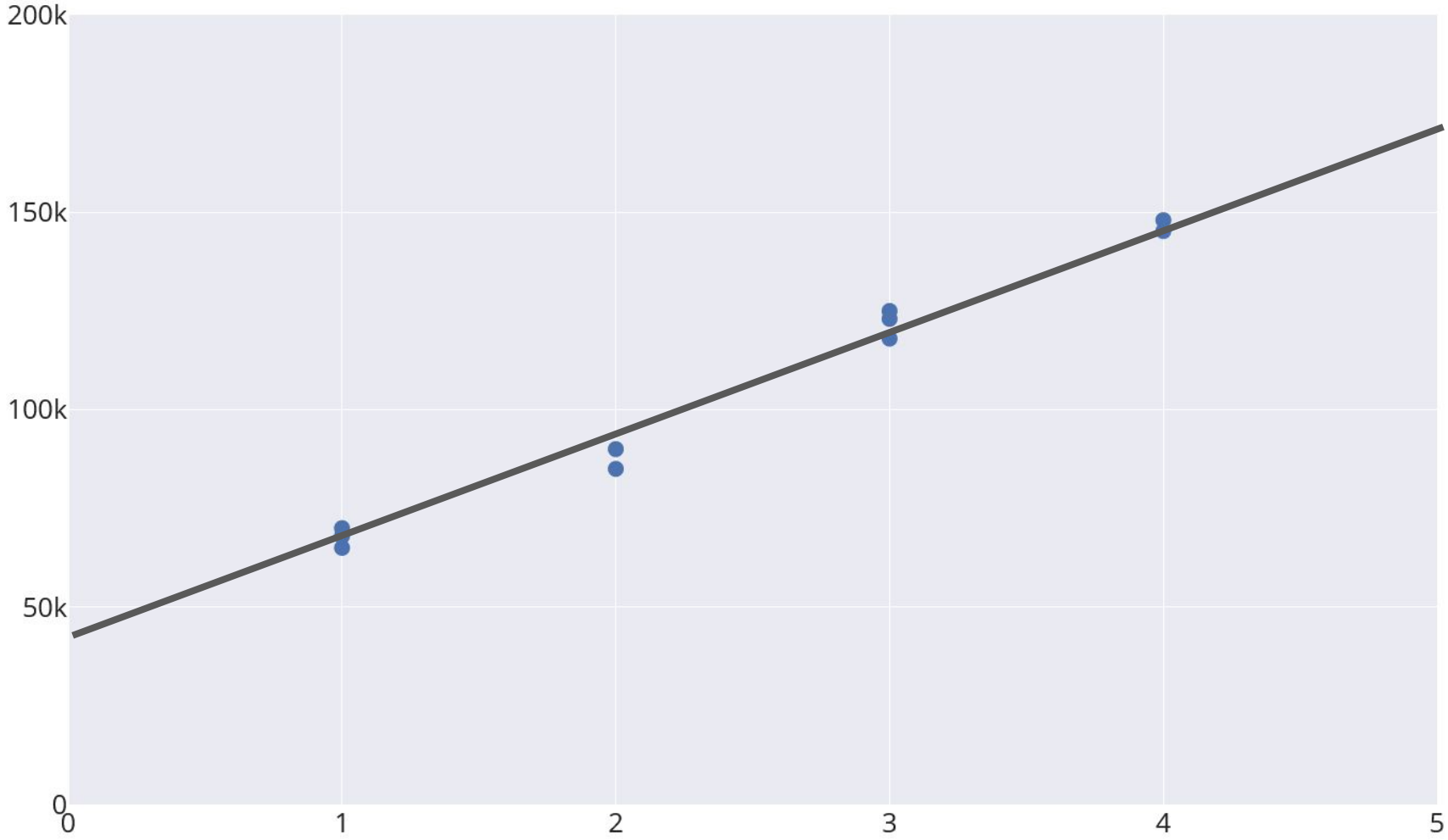
# Ex: Load Dataset

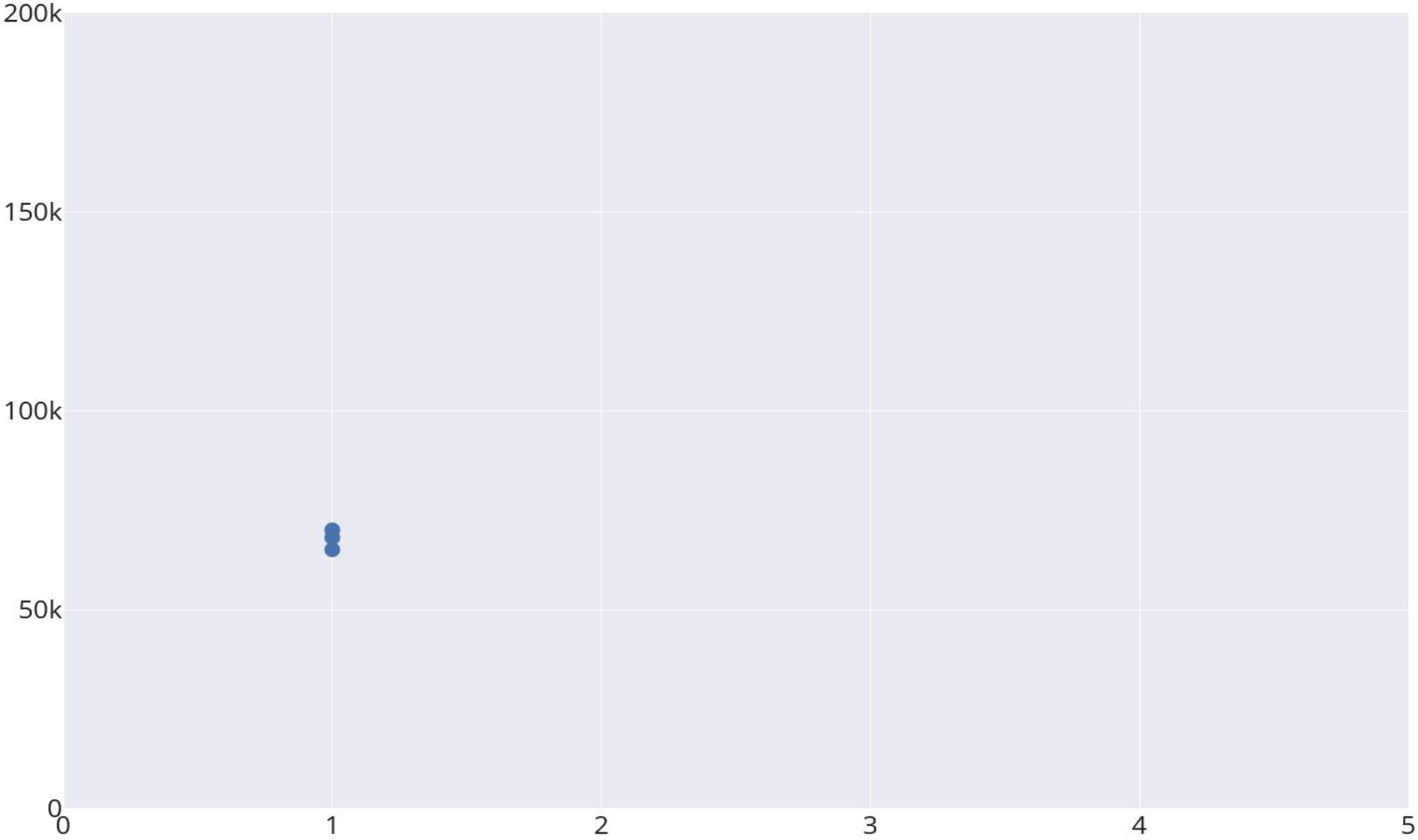
# df.shape quiz

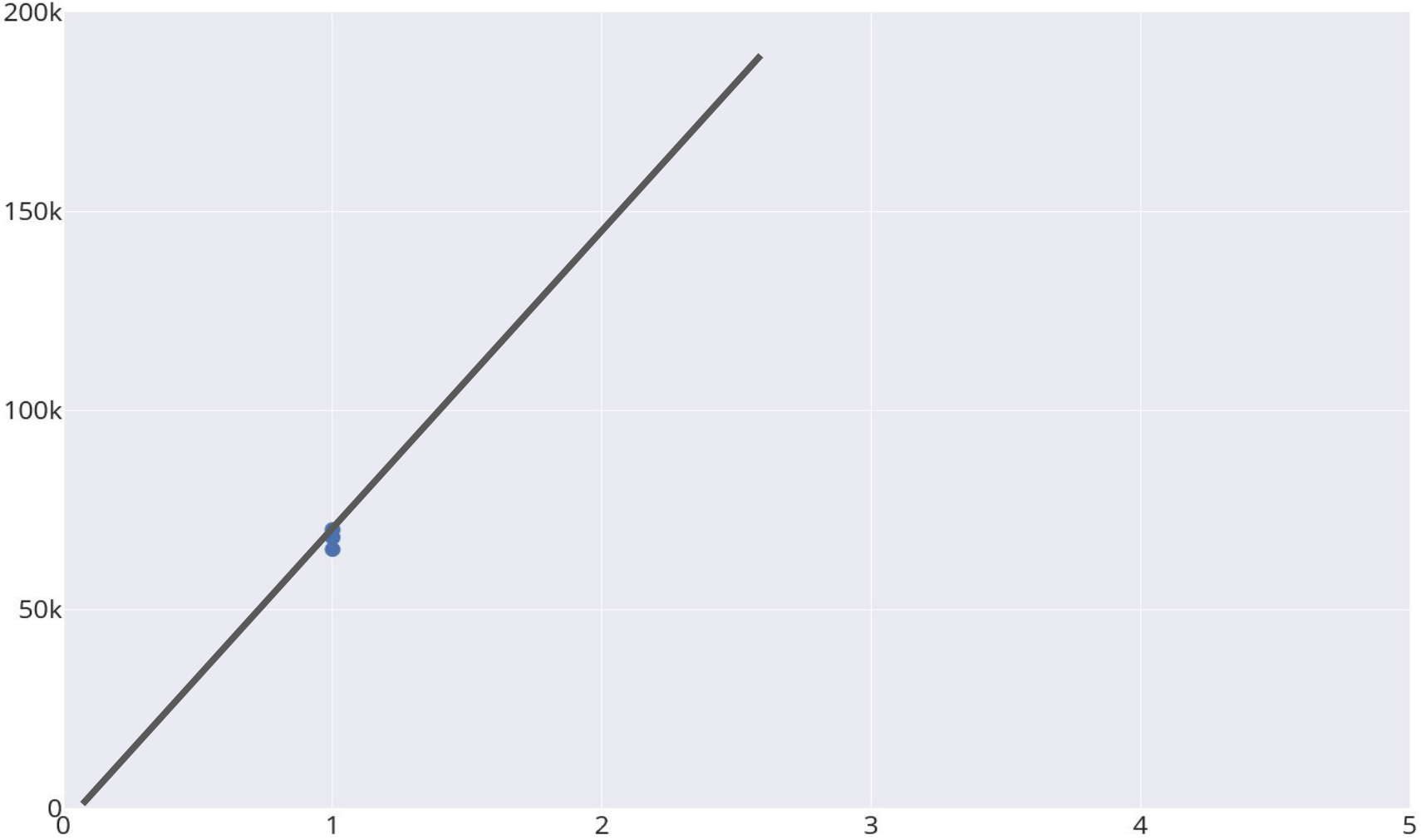


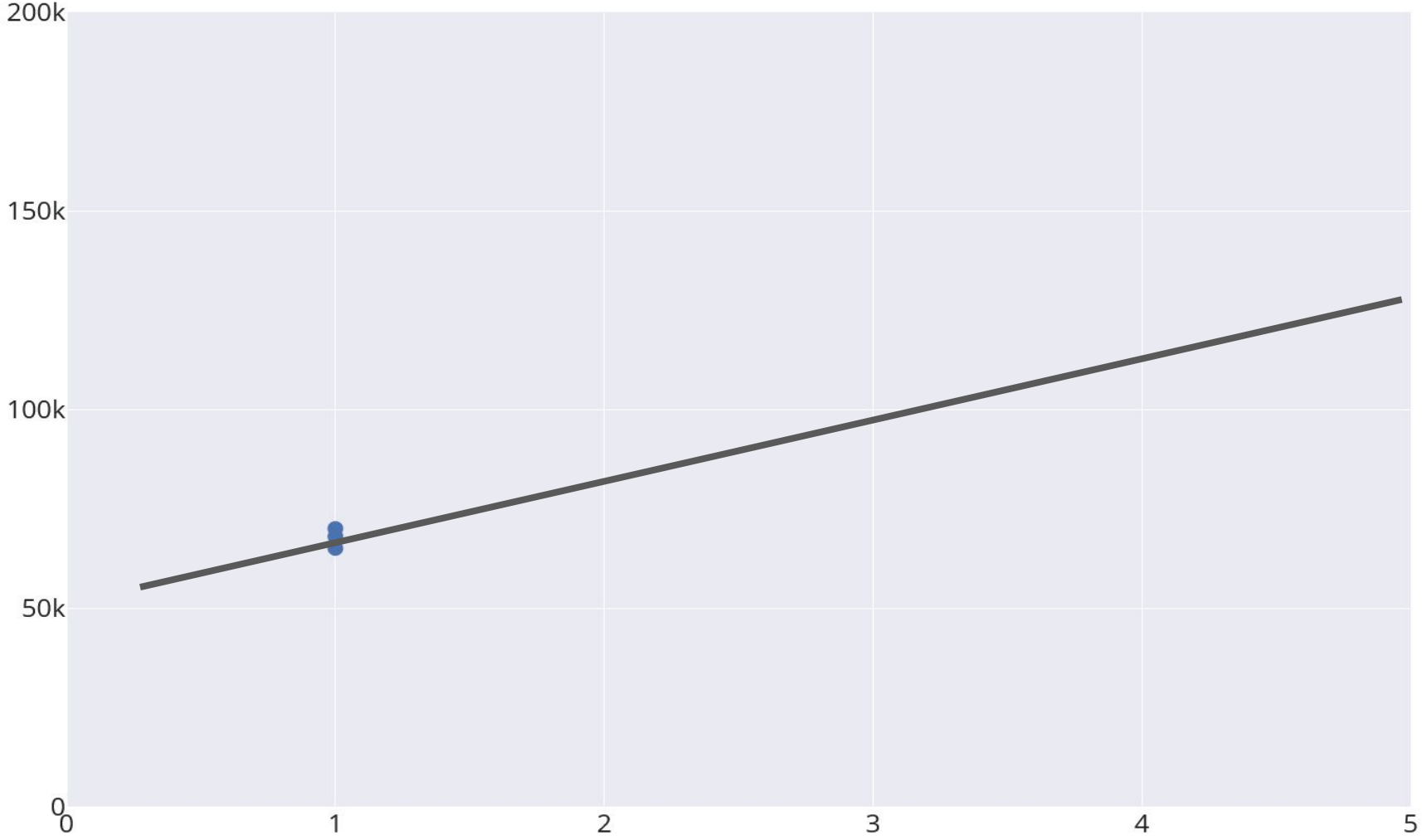
# GARBAGE IN GARBAGE OUT

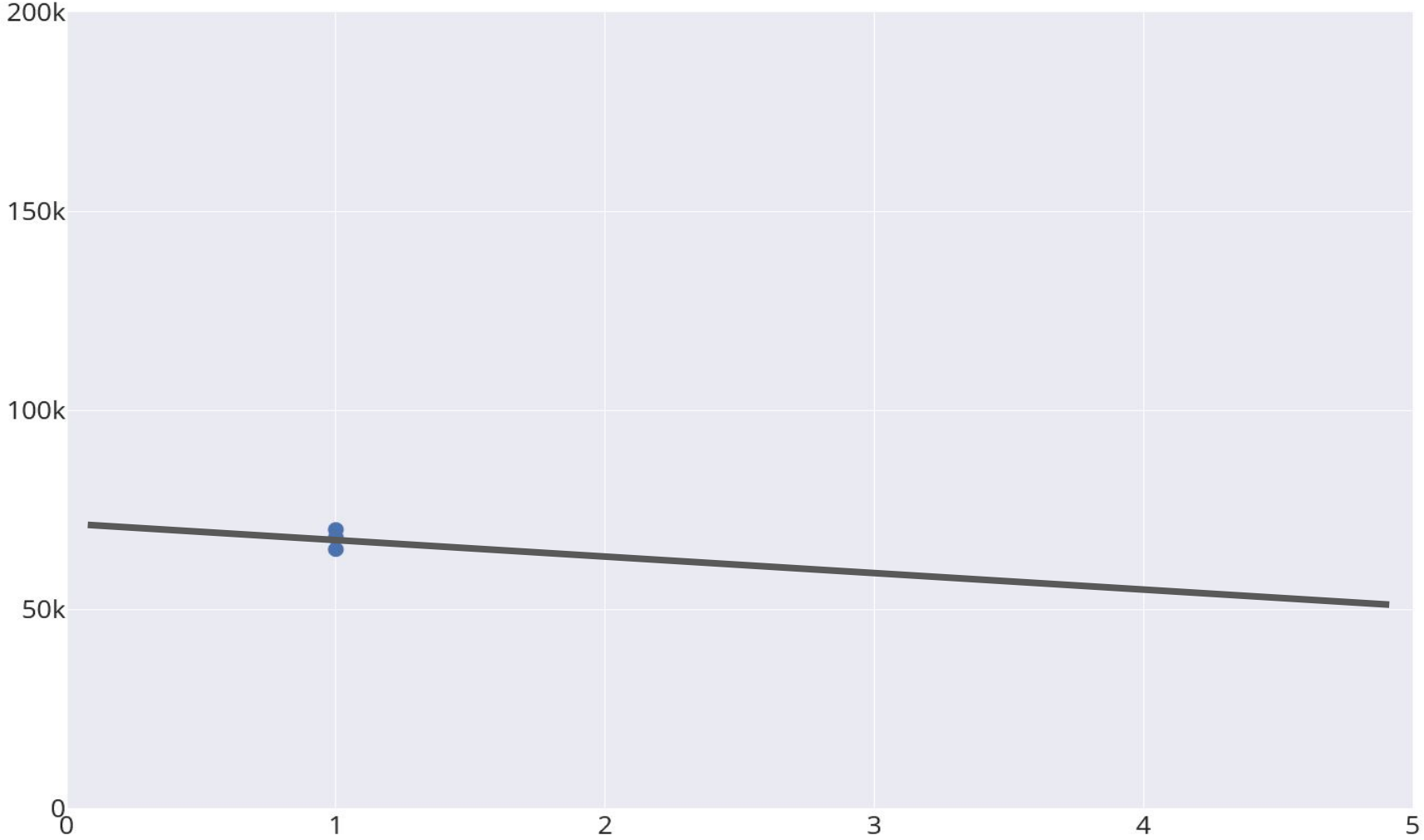

200k
150k
100k
50k
0
0
1
2
3
4
5


200k
150k
100k
50k
0
0
1
2
3
4
5


200k
150k
100k
50k
0
0
1
2
3
4
5


200k
150k
100k
50k
0
0
1
2
3
4
5


200k
150k
100k
50k
0
0
1
2
3
4
5


200k
150k
100k
50k
0
0
1
2
3
4
5

# Major Data Quality Issues

1. Duplicates
2. Missing Data
3. Incorrect Data
4. Inconsistent Formats
5. Insecure Data

# Duplicates

| | Street Name | Amount of Rooms | Area | Type | Price |
|---|---|---|---|---|---|
| **0** | Glacisstrasse | 5 | 220 | Luxury | 2500000 |
| **1** | Glacisstrasse | 5 | 220 | Luxury | 2500000 |
| **2** | Inffeldgasse | 3 | 120 | Standart | 450000 |
| **3** | Inffeldgasse | 3 | 120 | Standart | 450000 |
| **4** | Kasernstrasse | 2 | 45 | Econom | 100000 |
| **5** | Kasernstrasse | 2 | 45 | Econom | 100000 |
| **6** | Kreuzgasse | 6 | 250 | Luxury | 3000000 |
| **7** | Kreuzgasse | 6 | 250 | Luxury | 3000000 |

# How to deal with duplicates?

```
In [36]: df.drop_duplicates()
```

Out[36]:

| | Street Name | Amount of Rooms | Area | Type | Price |
|---|---|---|---|---|---|
| **0** | Glacisstrasse | 5 | 220 | Luxury | 2500000 |
| **2** | Inffeldgasse | 3 | 120 | Standart | 450000 |
| **4** | Kasernstrasse | 2 | 45 | Econom | 100000 |
| **6** | Kreuzgasse | 6 | 250 | Luxury | 3000000 |

# Ex. shape of the array after removing dub

# Missing Data

| | Street Name | Amount of Rooms | Area | Type | Price |
|---|---|---|---|---|---|
| **0** | Glacisstrasse | 5 | 220 | Luxury | NaN |
| **1** | Inffeldgasse | 3 | 120 | Standart | 450000.0 |
| **2** | NaN | 2 | 45 | Econom | NaN |
| **3** | Kreuzgasse | 6 | 250 | Luxury | 3000000.0 |

# Type of Missing Values

- Missing Completely at Random (MCAR)
- Missing at Random (MAR)
- Missing not at Random (MNAR)

# Missing Completely at Random (MCAR)

- Probability for a data point to be missing is completely random.
- There’s no relationship between whether a data point is missing and any values in the data set, missing or observed.
- The missing data are just a random subset of the data.

# Missing at Random (MAR)

- Probability for a data point to be missing is not related to the missing data, but it is related to some of the observed data.
- Example: ‘Whether or not someone answered #13 on your survey has nothing to do with the missing values, but it does have to do with the values of some other variable.’

# Missing not at Random (MNAR)

- There is a relationship between the probability of a value to be missing and its values
- Example: ‘Survey with regard to drug usage. Individuals being surveyed could potentially leave fields blank if they used drugs that are currently illegal out of fear of being prosecuted.’

# How to count Missing Data?

In [46]:
```
df
```

Out[46]:

| | Street Name | Amount of Rooms | Area | Type | Price |
|---|---|---|---|---|---|
| **0** | Glacisstrasse | 5 | 220 | Luxury | NaN |
| **1** | Inffeldgasse | 3 | 120 | Standart | 450000.0 |
| **2** | NaN | 2 | 45 | Econom | NaN |
| **3** | Kreuzgasse | 6 | 250 | Luxury | 3000000.0 |

In [49]:
```
df.Price.isna().sum()
```

Out[49]: 2

# How to remove Missing Data?

In [42]:
```
df.dropna()
```

Out[42]:

| | Street Name | Amount of Rooms | Area | Type | Price |
|---|---|---|---|---|---|
| **1** | Inffeldgasse | 3 | 120 | Standart | 450000.0 |
| **3** | Kreuzgasse | 6 | 250 | Luxury | 3000000.0 |

# SUM of missing values in column “X”

# Incorrect Data

| | Street Name | Amount of Rooms | Area | Type | Price |
|---|---|---|---|---|---|
| **0** | Glacisstrasse | -5 | 220 | Luxury | 2500000 |
| **1** | 1 | 3 | 120 | Standart | -1 |
| **2** | Kasernstrasse | 2020 | 45 | Econom | 100000 |
| **3** | Kreuzgasse | 3000000 | 250 | Luxury | 6 |

# Inconsistent Formats

| | Street Name | Amount of Rooms | Area | Type | Price |
|---|---|---|---|---|---|
| **0** | Glacisstrasse | 5 | 220 | Luxury | 2.500.000 |
| **1** | Inffeldgasse | 3 | 120 | Standart | 450000 Euro |
| **2** | Kasernstrasse | 2 | 45 | Econom | 100.000 eur |
| **3** | Kreuzgasse | 6 | 250 | Luxury | 3000000 € |

# Insecure Data

Data security & privacy laws are being put into place giving business extra financial incentive to follow these newly placed laws.

With steep fines for non-compliance, insecure data is quickly becoming one of the most dangerous types of dirty data.

# Insecure Data

| | Street Name | Amount of Rooms | Area | Type | Price | Religion of Owner |
|---|---|---|---|---|---|---|
| **0** | Glacisstrasse | 5 | 220 | Luxury | 2500000 | Christian |
| **1** | Inffeldgasse | 3 | 120 | Standart | 450000 | Muslim |
| **2** | Kasernstrasse | 2 | 45 | Econom | 100000 | Atheist |
| **3** | Kreuzgasse | 6 | 250 | Luxury | 3000000 | Jewish |

# How to deal with Insecure Data?

```
In [54]: df.drop(labels='Religion of Owner', axis=1)
```

Out[54]:

| | Street Name | Amount of Rooms | Area | Type | Price |
|---|---|---|---|---|---|
| **0** | Glacisstrasse | 5 | 220 | Luxury | 2500000 |
| **1** | Inffeldgasse | 3 | 120 | Standart | 450000 |
| **2** | Kasernstrasse | 2 | 45 | Econom | 100000 |
| **3** | Kreuzgasse | 6 | 250 | Luxury | 3000000 |

# DROP NAME

# Most common data types?

| dtypes | Example |
|---|---|
| float | 3.14, 5.16, 0.1111 |
| int | 23, 111, 45, 69 |
| datetime | 2019.01.01 30.03.01 |
| timedelta | 23:59 |
| Strings | ‘This’, ‘course’, ‘is amazing’ |
| bool | True, False |

# Exploratory Data Analysis

# What is EDA?

Cleaning your data
Summarizing your data
Finding patterns

# What is EDA?

Cleaning your data
Summarizing your data
Finding patterns
Telling a story with you data.

# Why EDA?

Understand the quality of your data.
Gain some quick insights.
Find potential patterns.

# Descriptive Statistics

# Descriptive Statistics

The art of describing the basic features of your data.

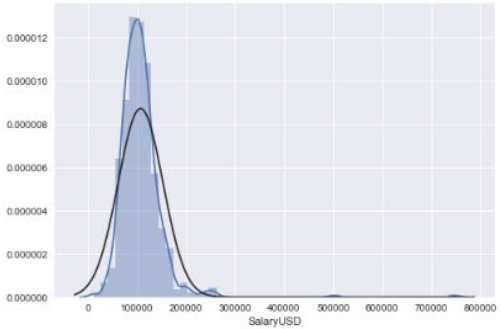
The basis of all quantitative analysis of data.

# Univariate Analysis

| Salary |
|---|
| 54.000€ |
| 90.000€ |
| 67.000€ |
| 96.000€ |
| -10.000€ |
| 67.000€ |

**MAX:** 9.000€
**MEAN:** 79.560€
**MIN:** 54.000€

**Why is there a salary of -10.000€?**



# Technique #1 - Missing values?

Determine how many values are missing.

| Salary |
| --- |
| 54.000€ |
| **?** |
| 67.000€ |
| 96.000€ |
| **?** |
| 67.000€ |

**2 Missing values**

Sometimes missing values, are not obvious

| JOB |
| --- |
| Data Scientist |
| **Not Known** |
| Data Scientist |
| Potato Scientist |
| **Not Known** |
| **Not Known** |

**3 Missing values**

# Technique #2 - Distribution of the data

Describe your data with basic statistical functions

| Salary |
| --- |
| 54.000€ |
| 65.000€ |
| 67.000€ |
| 96.000€ |
| 79.000€ |
| 67.000€ |

**MAX:** 96.000€
**MEAN:** 79.560€
**MEDIAN:** 79.560€
**MIN:** 54.000€
**STD:** 15.000€

# What is the mean of this table?

| Salary |
| --- |
| Data Scientist |
| Data Scientist |
| Data Scientist |
| Potato Farmer |
| Potato Farmer |
| Potato Farmer |

???

# What is the mean of this table?

Categorical data statistics are different.

| Salary |
| --- |
| Data Scientist |
| Data Scientist |
| Data Scientist |
| Potato Farmer |
| Potato Farmer |
| Potato Farmer |

???

**Farmer Scientist?**

# What is the mean of this table?

Categorical data statistics are different.

| Salary |
| --- |
| Data Scientist |
| Data Scientist |
| Data Scientist |
| Potato Farmer |
| Potato Farmer |
| Potato Farmer |

???

**Potato Scientist?**

Categorical data statistics are different.

| JOB |
| --- |
| Data Scientist |
| Data Engineer |
| Data Scientist |
| Potato Scientist |
| Data Engineer |
| Data Engineer |

**Unique Values:**
Data Scientist
Potato Scientist
Data Engineer

Categorical data statistics are different.

| JOB |
| --- |
| Data Scientist |
| Data Engineer |
| Data Scientist |
| Potato Scientist |
| Data Engineer |
| Data Engineer |

**Number of Unique Values: 3**

Categorical data statistics are different.

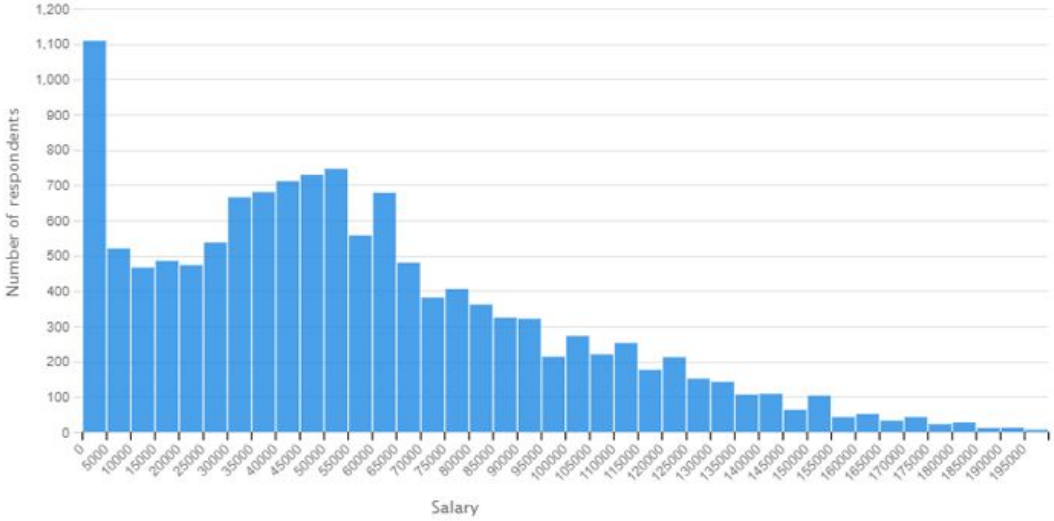
| JOB |
| --- |
| Data Scientist |
| Data Engineer |
| Data Scientist |
| Potato Scientist |
| Data Engineer |
| Data Engineer |

**Value Counts:**
Data Scientist: 3
Data Engineer: 3
Potato Scientist: 1

# Technique #3 - Histogram Plot

Visualizing the shape tells you more about the data.

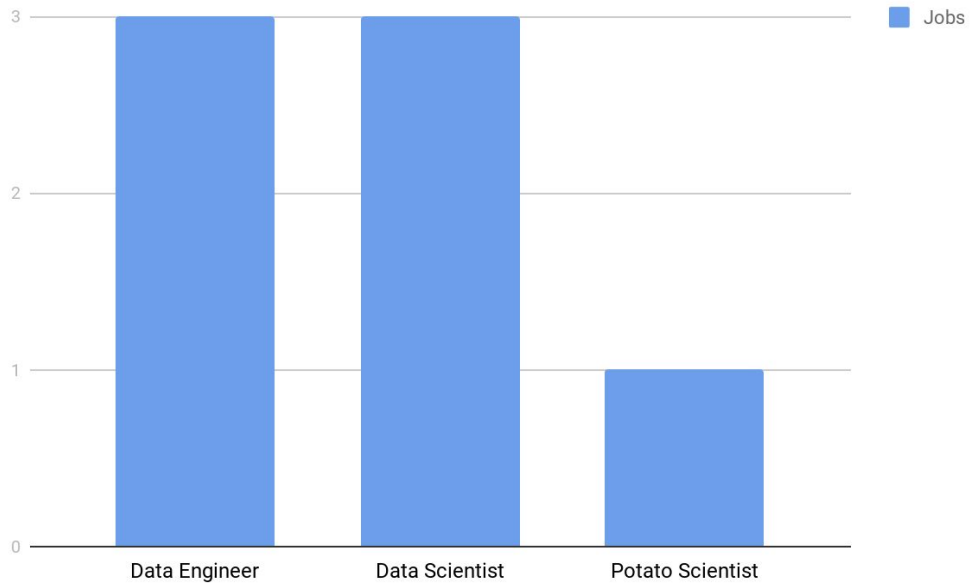
| Salary |
| --- |
| 54.000€ |
| 56.425€ |
| 67.000€ |
| 96.000€ |
| 69.420€ |
| 67.000€ |



# Technique #4 - Count Plot

Visualizing the shape tells you more about the data.

| JOB |
| --- |
| Data Scientist |
| Data Engineer |
| Data Scientist |
| Potato Scientist |
| Data Engineer |
| Data Engineer |



# Technique #5 - Identifying Extreme Points

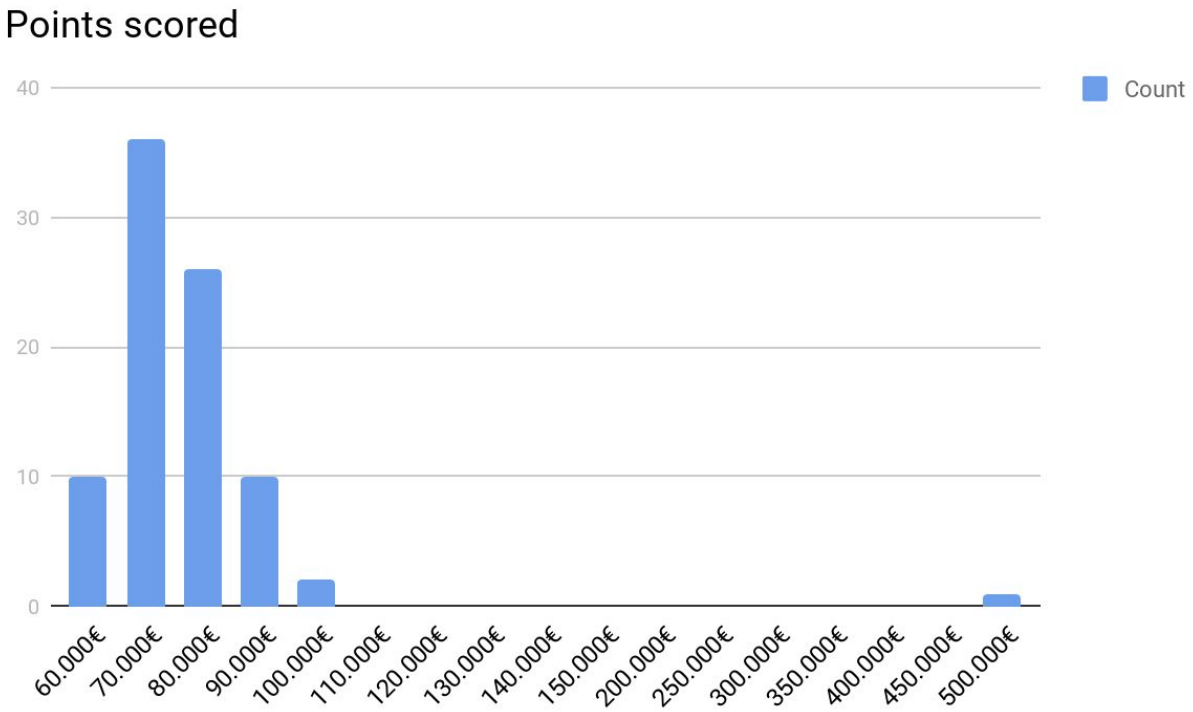
Some data points stick out.

| SALARY |
| --- |
| 70.000€ |
| 74.000€ |
| 79.000€ |
| 69.000€ |
| 65.000€ |
| **500.000€** |

**The median salary is 74.000€?**
500.000€ is way over the median.

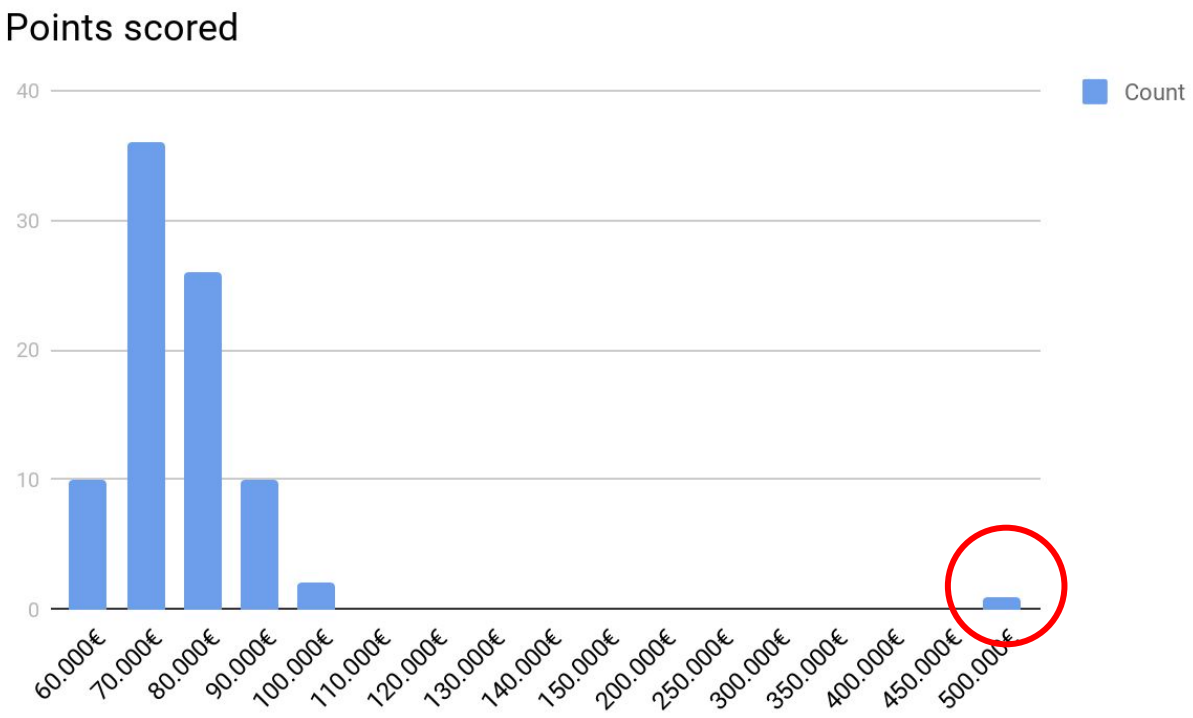
We can use the previous techniques to identify these points as well.

| SALARY |
| --- |
| 70.000€ |
| 74.000€ |
| 79.000€ |
| 69.000€ |
| 65.000€ |
| **500.000€** |



We can use the previous techniques to identify these points.

| SALARY |
| --- |
| 70.000€ |
| 74.000€ |
| 79.000€ |
| 69.000€ |
| 65.000€ |
| **500.000€** |



# Why is the salary so high?

It is hard to answer questions this question with one variable only.

# Bivariate Analysis

| SALARY | JOB |
| --- | --- |
| 70.000€ | Data Scientist |
| 74.000€ | Data Scientist |
| 79.000€ | Data Engineer |
| 69.000€ | Data Engineer |
| 65.000€ | Data Analyst |
| | |

| SALARY | JOB |
| --- | --- |
| 70.000€ | Data Scientist |
| 74.000€ | Data Scientist |
| 79.000€ | Data Engineer |
| 69.000€ | Data Engineer |
| 65.000€ | Data Analyst |
| **500.000€** | **CEO** |

| | | | |
|---|---|---|---|
| 65.000€ | Data Scientist | 65.000€ | Data Scientist |
| 45.000€ | Manager | 45.000€ | Manager |
| 50.000€ | Data Engineer | 50.000€ | Data Engineer |
| 75.000€ | Analyst | 75.000€ | Analyst |
| 85.000€ | Controller | 85.000€ | Controller |
| 50.000€ | Chess Player | 50.000€ | Chess Player |
| 68.667€ | Trucker | 68.667€ | Trucker |
| 70.667€ | Data Scientist | 70.667€ | Data Scientist |
| 72.667€ | Manager | 72.667€ | Manager |
| 74.667€ | Data Engineer | 74.667€ | Data Engineer |
| 76.667€ | Analyst | 76.667€ | Analyst |
| 78.667€ | Controller | 78.667€ | Controller |
| 80.667€ | Chess Player | 80.667€ | Chess Player |
| 82.667€ | Trucker | 82.667€ | Trucker |
| 84.667€ | Data Scientist | 84.667€ | Data Scientist |
| 86.667€ | Manager | 86.667€ | Manager |
| 88.667€ | Data Engineer | 88.667€ | Data Engineer |
| 90.667€ | Analyst | 90.667€ | Analyst |
| 92.667€ | Controller | 92.667€ | Controller |
| 94.667€ | Chess Player | 94.667€ | Chess Player |
| 96.667€ | Trucker | 96.667€ | Trucker |
| 98.667€ | Data Scientist | 98.667€ | Data Scientist |
| 100.667€ | Data Scientist | 100.667€ | Manager |
| 102.667€ | Data Scientist | 500.000€ | CEO |
| 104.667€ | Data Scientist | 104.667€ | Analyst |
| 106.667€ | Data Scientist | 106.667€ | Controller |

| | | | |
|---|---|---|---|
| 65.000€ | Data Scientist | 65.000€ | Data Scientist |
| 45.000€ | Manager | 45.000€ | Manager |
| 50.000€ | Data Engineer | 50.000€ | Data Engineer |
| 75.000€ | Analyst | 75.000€ | Analyst |
| 85.000€ | Controller | 85.000€ | Controller |
| 50.000€ | Chess Player | 50.000€ | Chess Player |
| 68.667€ | Trucker | 68.667€ | Trucker |
| 70.667€ | Data Scientist | 70.667€ | Data Scientist |
| 72.667€ | Manager | 72.667€ | Manager |
| 74.667€ | Data Engineer | 74.667€ | Data Engineer |
| 76.667€ | Analyst | 76.667€ | Analyst |
| 78.667€ | Controller | 78.667€ | Controller |
| 80.667€ | Chess Player | 80.667€ | Chess Player |
| 82.667€ | Trucker | 82.667€ | Trucker |
| 84.667€ | Data Scientist | 84.667€ | Data Scientist |
| 86.667€ | Manager | 86.667€ | Manager |
| 88.667€ | Data Engineer | 88.667€ | Data Engineer |
| 90.667€ | Analyst | 90.667€ | Analyst |
| 92.667€ | Controller | 92.667€ | Controller |
| 94.667€ | Chess Player | 94.667€ | Chess Player |
| 96.667€ | Trucker | 96.667€ | Trucker |
| 98.667€ | Data Scientist | 98.667€ | Data Scientist |
| 100.667€ | Data Scientist | 100.667€ | Manager |
| 102.667€ | Data Scientist | 500.000€ | CEO |
| 104.667€ | Data Scientist | 104.667€ | Analyst |
| 106.667€ | Data Scientist | 106.667€ | Controller |

# Technique #6 - Compare Statistics

We can group previously analyzed statistics.

| SALARY | JOB |
|---|---|
| 70.000€ | Data Scientist |
| 74.000€ | Data Scientist |
| 79.000€ | Data Engineer |
| 69.000€ | Data Engineer |
| 65.000€ | Data Analyst |
| **500.000€** | **CEO** |

# Technique #6 - Compare Statistics

We can group previously analyzed statistics.

| SALARY | JOB |
|---|---|
| 70.000€ | Data Scientist |
| 74.000€ | Data Scientist |
| 79.000€ | Data Engineer |
| 69.000€ | Data Engineer |
| 65.000€ | Data Analyst |
| **500.000€** | **CEO** |

**Data Scientist**
**MAX:** 74.000€
**MEAN:** 72.000€

**CEO**
**MAX:** 500.000€
**MEAN:** 500.000€

# Technique #7 - Scatterplot

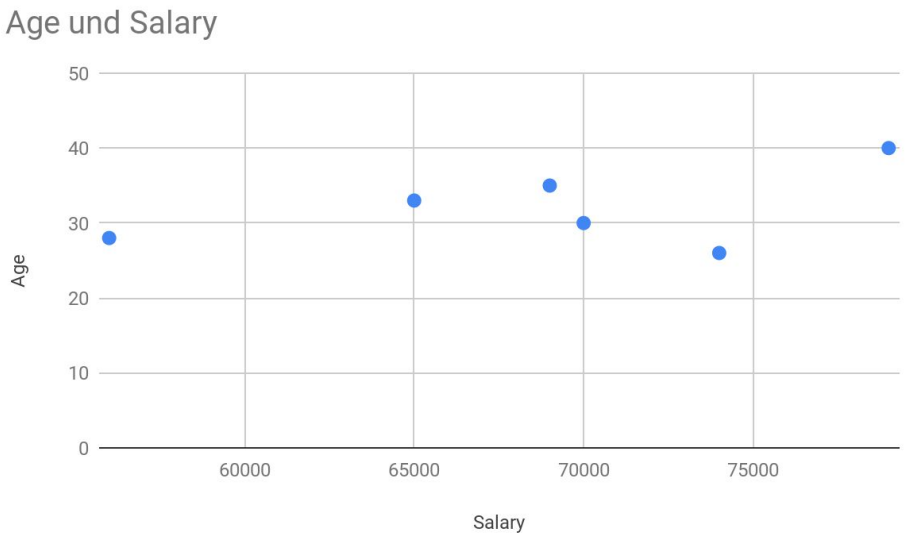
Great way to compare visually 2 continuous variables

| SALARY | Age |
|---|---|
| 70.000€ | 30 |
| 74.000€ | 26 |
| 79.000€ | 40 |
| 69.000€ | 35 |
| 65.000€ | 33 |
| 56.000€ | 28 |

# Technique #7 - Scatterplot

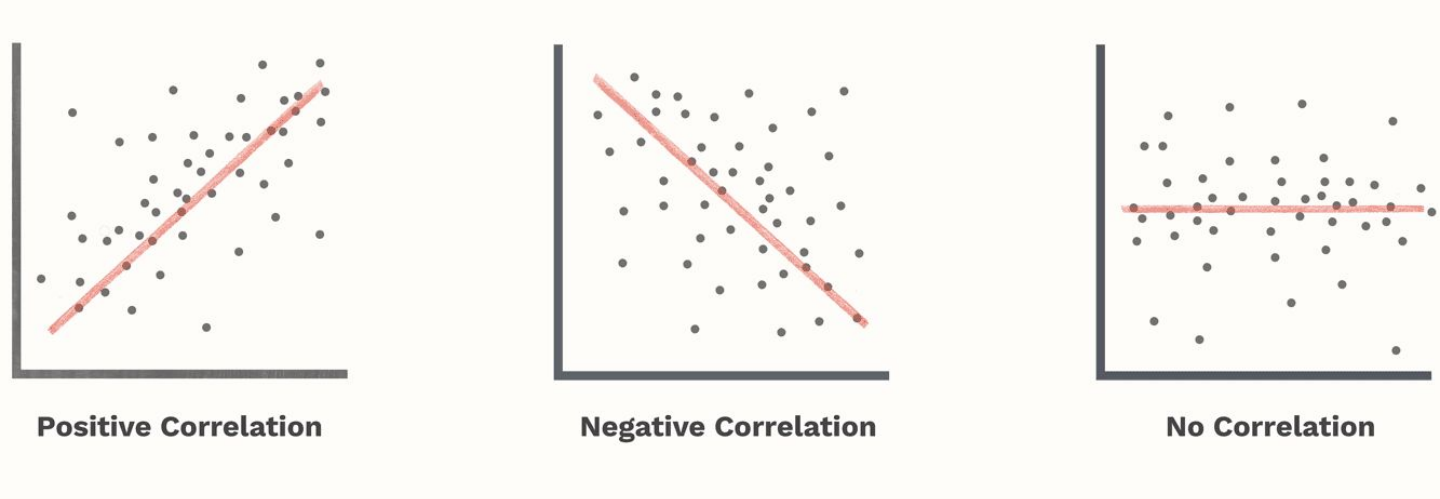
Great way to compare visually 2 continuous variables

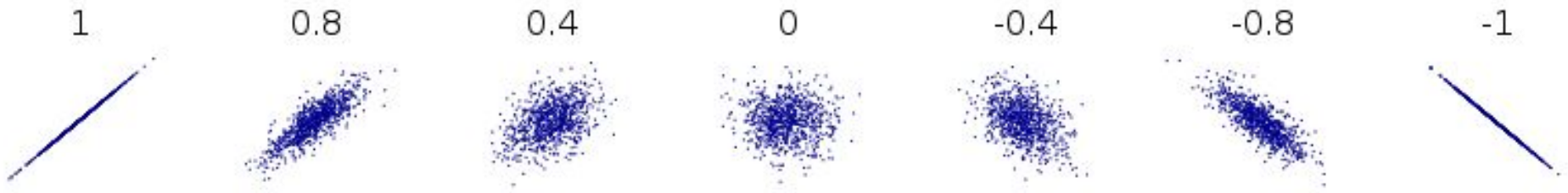
| SALARY | Age |
|---|---|
| 70.000€ | 30 |
| 74.000€ | 26 |
| 79.000€ | 40 |
| 69.000€ | 35 |
| 65.000€ | 33 |
| 56.000€ | 28 |



# Technique #8 - Linear Correlation

Linear correlation refers to straight-line relationships between two variables.




1
0.8
0.4
0
-0.4
-0.8
-1

**Ok, so what.**
**What does this help me?**

## Ok, so what.
## What does this help me?

Correlation measures the extent to which variables:

- depend on one another
- predict one another

# Rainfall is positively correlated with amount of vegetation

**Pollen count is positively correlated with bee activity.**

# Pollen count is positively correlated with bee activity.

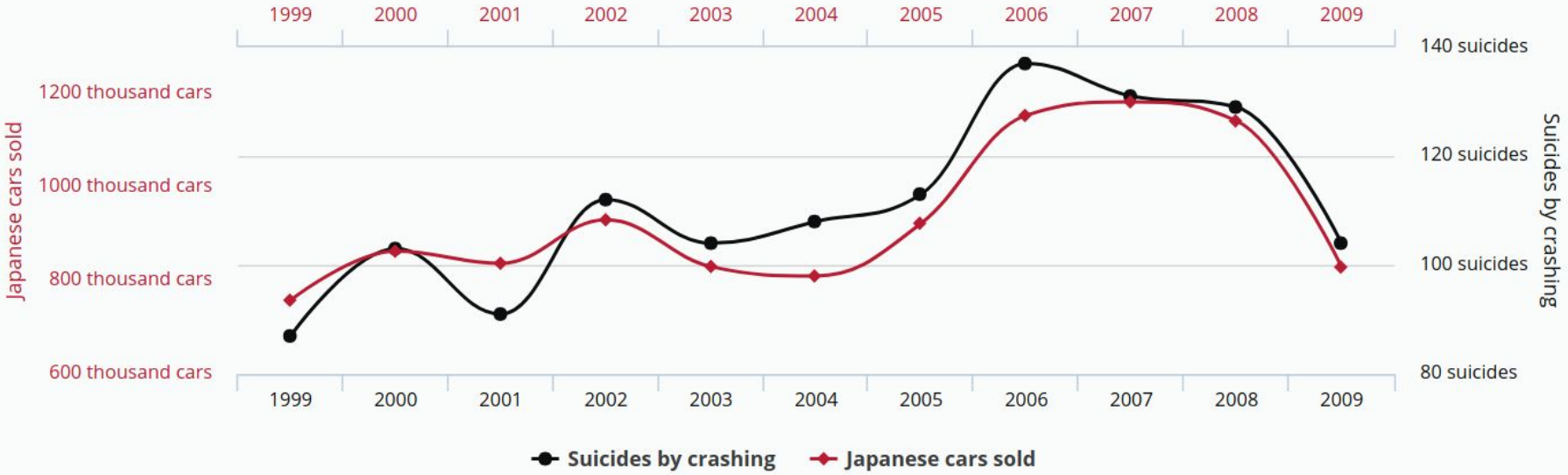
**Is bee activity the cause of the pollen count?**

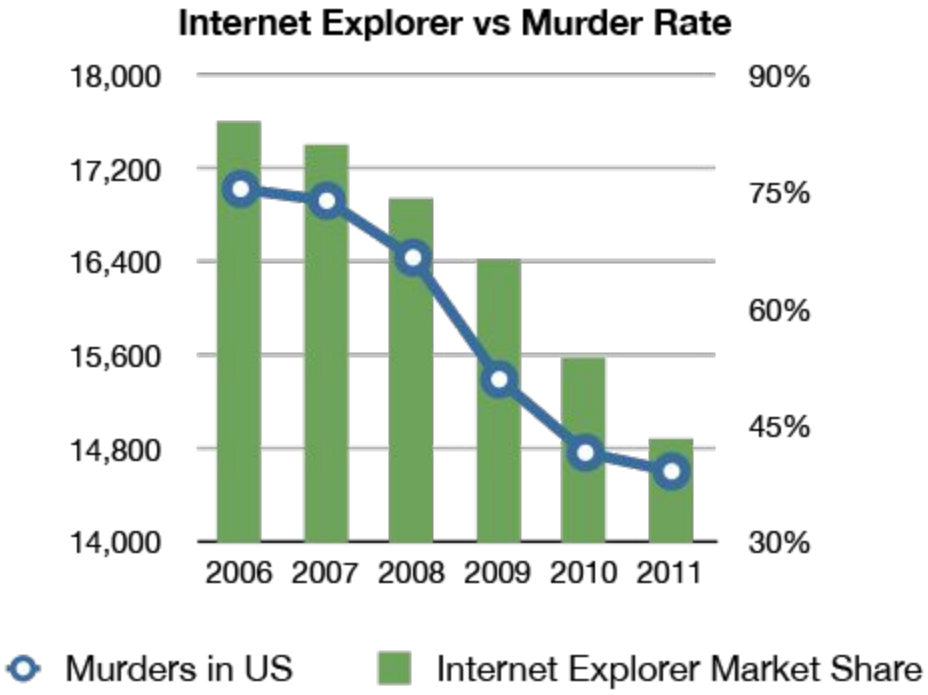
# Japanese passenger cars sold in the US

correlates with

# Suicides by crashing of motor vehicle

Correlation: 93.57% (r=0.935701)




Internet Explorer vs Murder Rate
18,000
17,200
16,400
15,600
14,800
14,000
90%
75%
60%
45%
30%
2006
2007
2008
2009
2010
2011
Murders in US
Internet Explorer Market Share