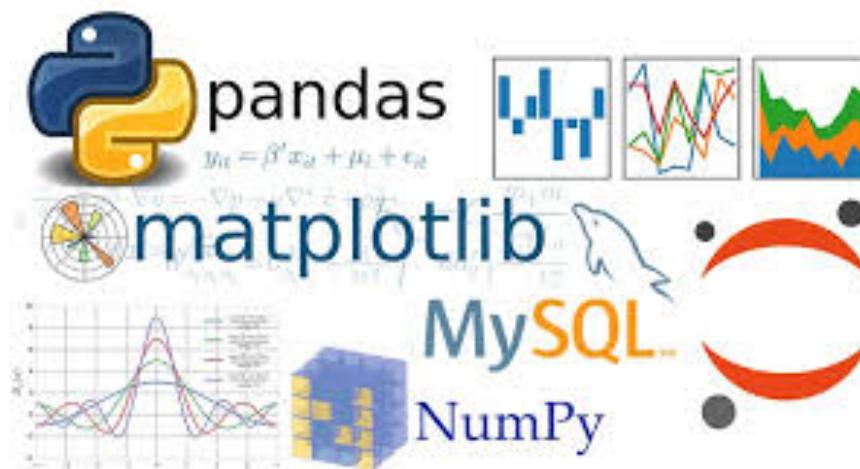
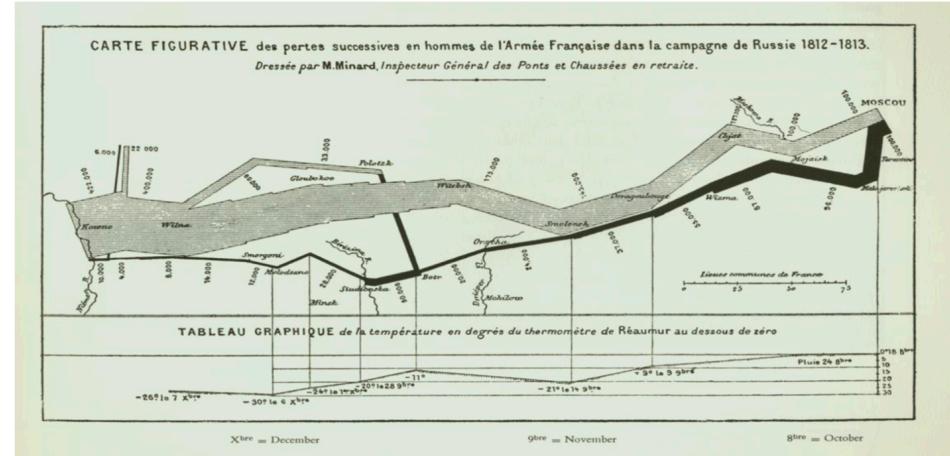
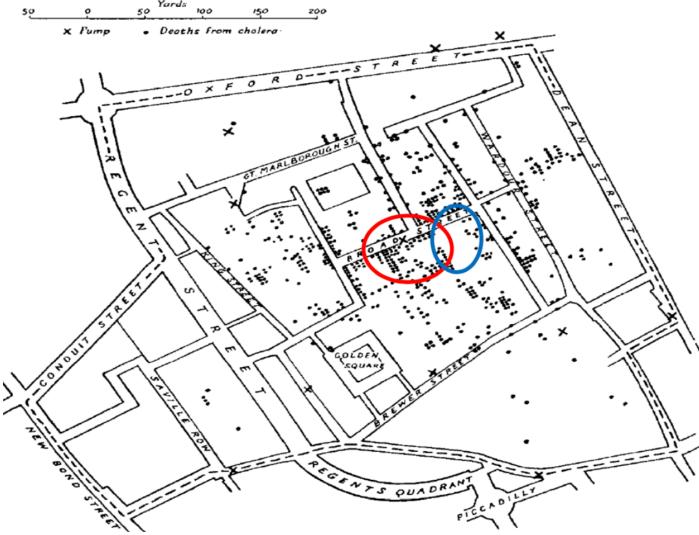


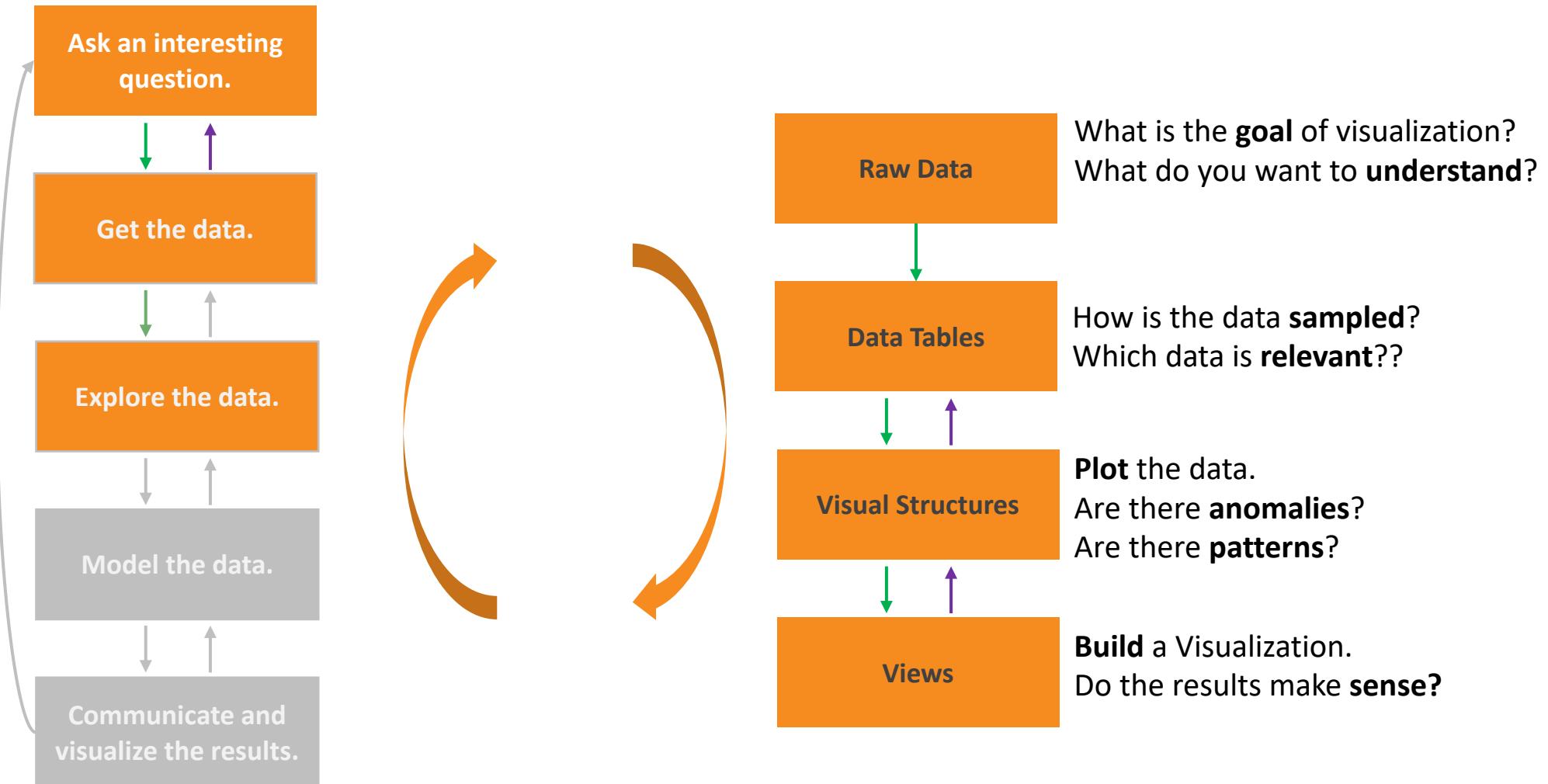
Data Visualization

Benjamin M. Abdel-Karim

Motivation



Get the Data and Explore the Data



Our Data set

The screenshot shows the Kaggle website interface. At the top, there is a cookie consent banner with options to "Got it" or "Learn more". Below the banner, the navigation bar includes the "kaggle" logo, a search bar with a magnifying glass icon, and links for "Competitions", "Datasets", "Notebooks", "Discussion", "Courses", and a "Sign in" button. A "Register" button is also visible. The main content area displays a dataset titled "Lending Club Loan Data", which is described as "Analyze Lending Club's issued loans". The dataset was created by Wendy Kan and updated 6 months ago (Version 1). It has 1080 rows and 2 GB of data. Below the title, there are tabs for "Data" (which is selected), "Kernels (587)", "Discussion (34)", "Activity", and "Metadata". There are also buttons for "Download (2 GB)" and "New Notebook". Further down, there are sections for "Usability" (rating 5.9) and "Tags" (finance, loans). The "Description" section contains a detailed text about the dataset, stating that it contains complete loan data for all loans issued through 2007-2015, including current status, payment information, and additional features like credit scores and address details.

We use cookies on kaggle to deliver our services, analyze web traffic, and improve your experience on the site. By using kaggle, you agree to our use of cookies.

Got it Learn more

kaggle Search Competitions Datasets Notebooks Discussion Courses ... Sign in Register

Dataset

Lending Club Loan Data

Analyze Lending Club's issued loans

Wendy Kan • updated 6 months ago (Version 1)

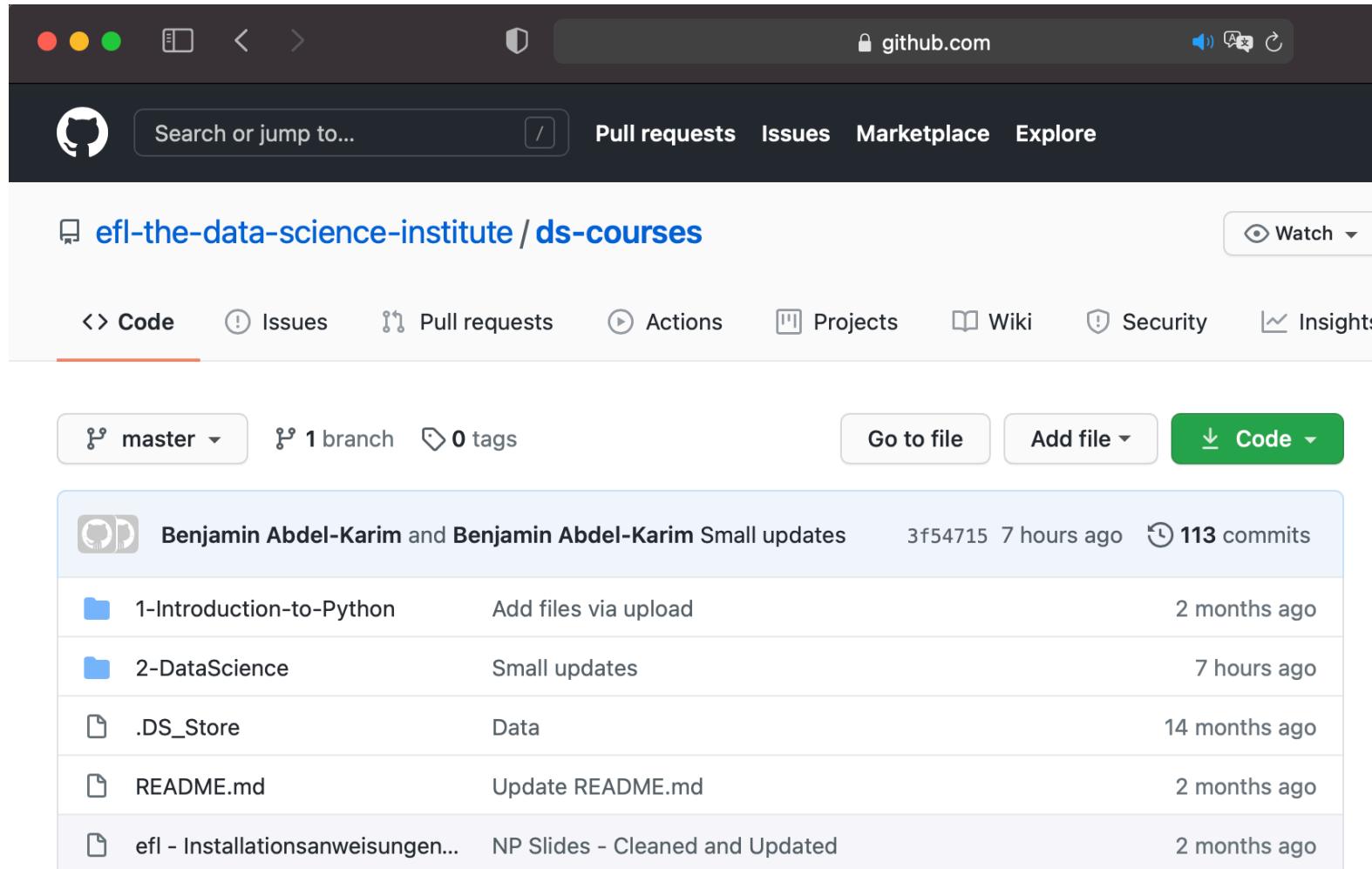
Data Kernels (587) Discussion (34) Activity Metadata Download (2 GB) New Notebook

Usability 5.9 Tags finance, loans

Description

These files contain complete loan data for all loans issued through the 2007-2015, including the current loan status (Current, Late, Fully Paid, etc.) and latest payment information. The file containing loan data through the "present" contains complete loan data for all loans issued through the previous completed calendar quarter. Additional features include credit scores, number of finance inquiries, address including zip codes, and state, and collections among others. The file is a matrix of about 890 thousand observations and 75 variables. A data dictionary is provided in a separate file. k

Materials



The screenshot shows a GitHub repository page. At the top, there's a navigation bar with icons for window control, a shield, a lock, and a search bar containing 'github.com'. Below the bar are links for 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. The main title is 'efl-the-data-science-institute / ds-courses'. To the right is a 'Watch' button. Below the title, there are tabs for 'Code', 'Issues', 'Pull requests', 'Actions', 'Projects', 'Wiki', 'Security', and 'Insights'. The 'Code' tab is selected. Underneath, there's a dropdown for the 'master' branch, showing '1 branch' and '0 tags'. There are buttons for 'Go to file', 'Add file', and a green 'Code' button. A commit summary shows 'Benjamin Abdel-Karim and Benjamin Abdel-Karim' made 'Small updates' at '3f54715' 7 hours ago, with '113 commits'. Below this, a list of files and their details is shown:

File	Description	Last Updated
1-Introduction-to-Python	Add files via upload	2 months ago
2-DataScience	Small updates	7 hours ago
.DS_Store	Data	14 months ago
README.md	Update README.md	2 months ago
efl - Installationsanweisungen...	NP Slides - Cleaned and Updated	2 months ago

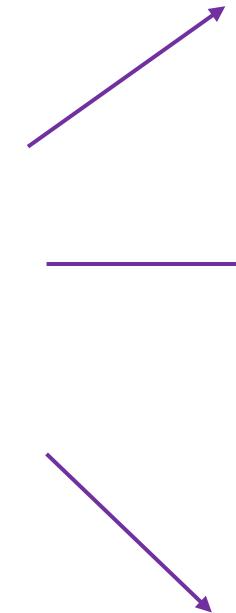


<https://github.com/efl-the-data-science-institute/ds-courses>

Why are insides so important?



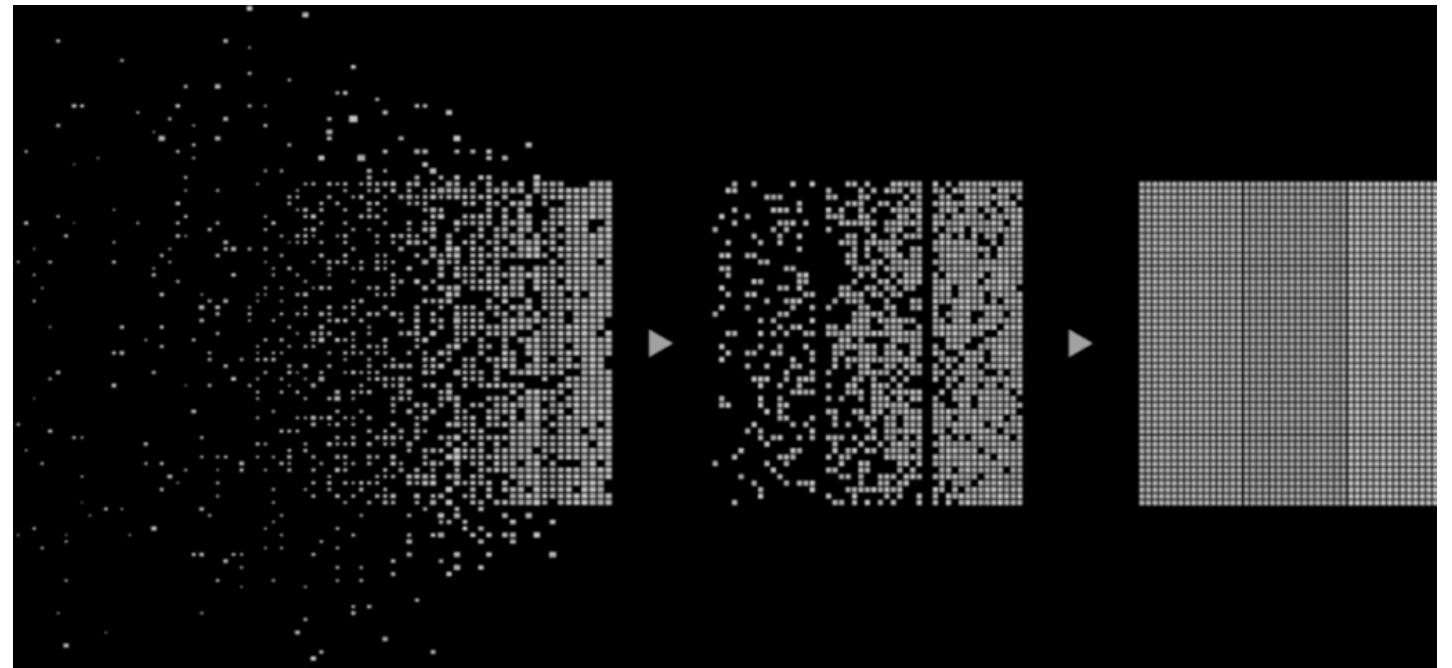
auxmoney
Etc.



Investor Perspective



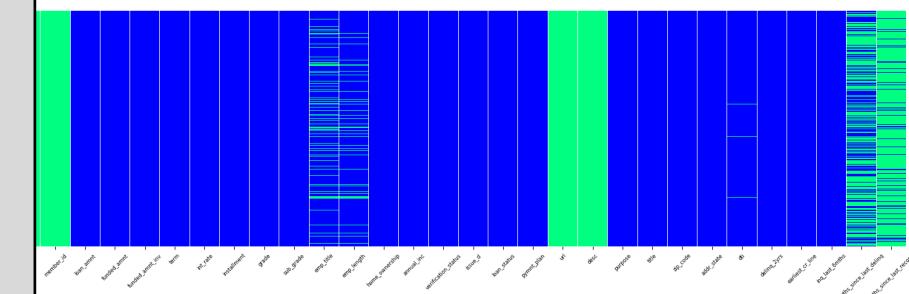
Step 1 – Pre-processing



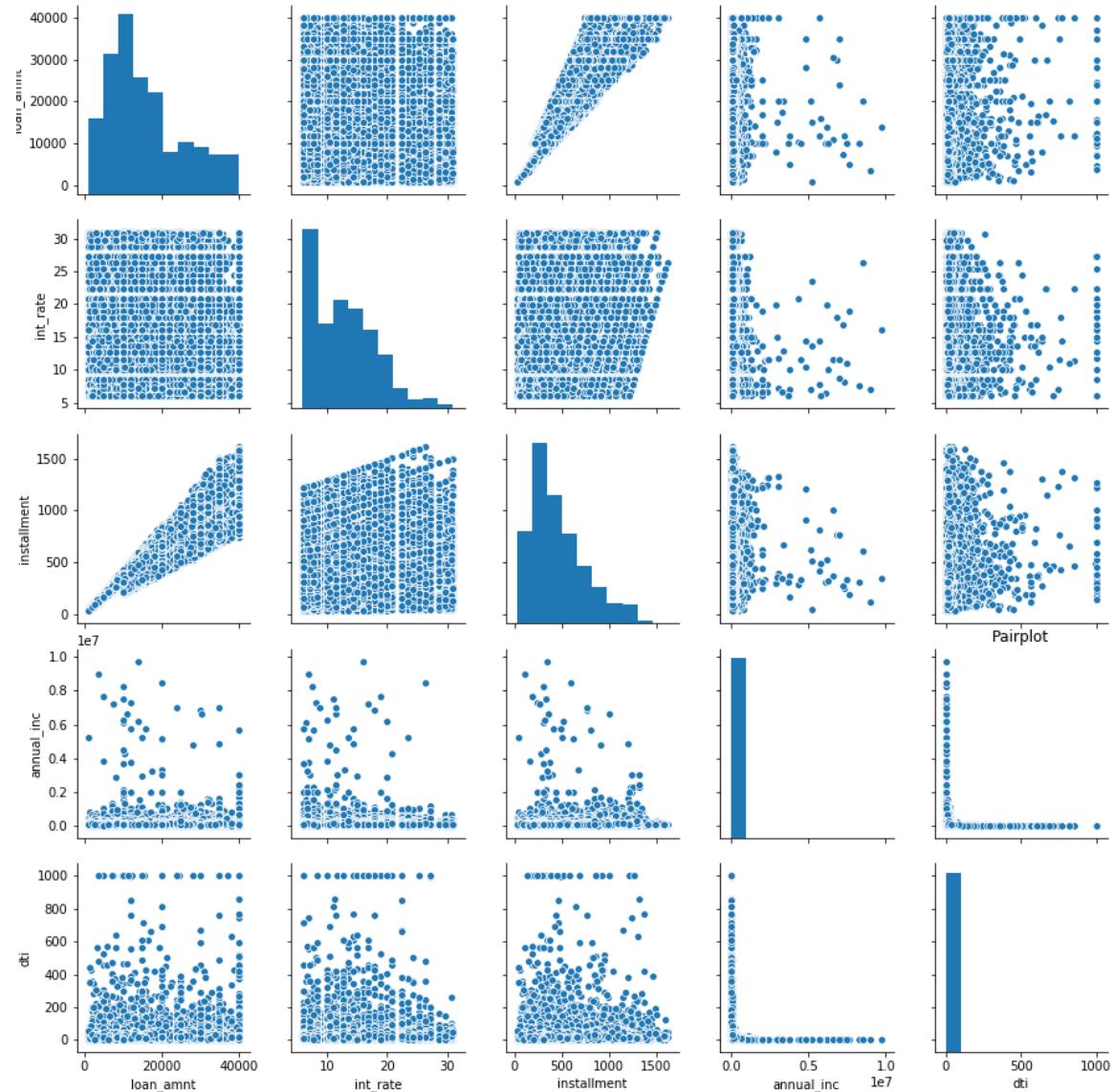
Pre-Processing

- Data pre-processing is an important step in the data mining process. The phrase "garbage in, garbage out" is particularly applicable to data mining and machine learning projects.
- Find all missing values.
- We should get an overview of the missing values.

```
plt.figure(figsize=(15, 5))
sns.heatmap(df.isnull(), cbar=False,
yticklabels=False, cmap='winter')
plt.xticks(rotation=45, fontsize=6)
plt.tight_layout()
plt.savefig('figures/MissingValues.png')
plt.show()
plt.close()
```

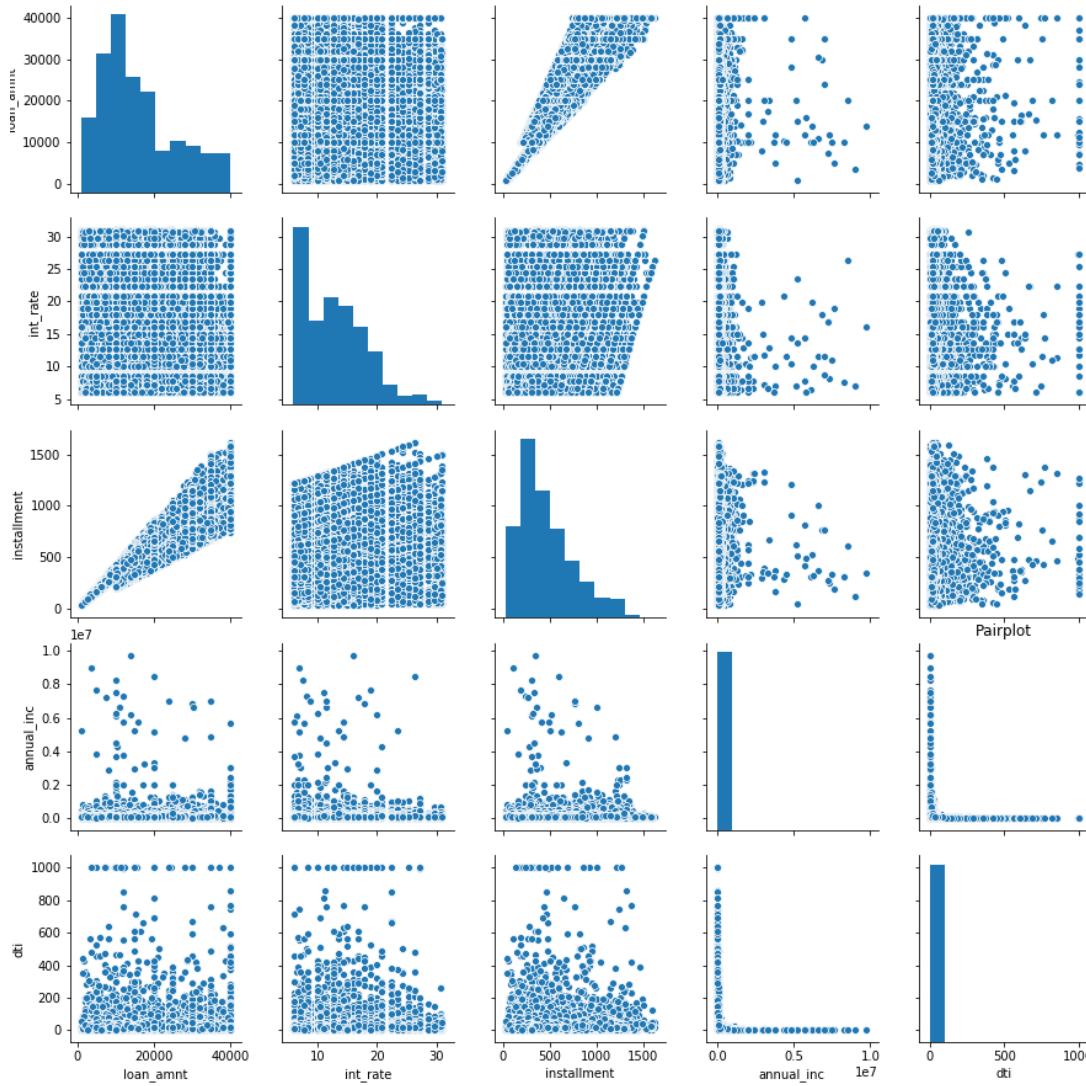


Get an
Overview
(Feeling) of
the data



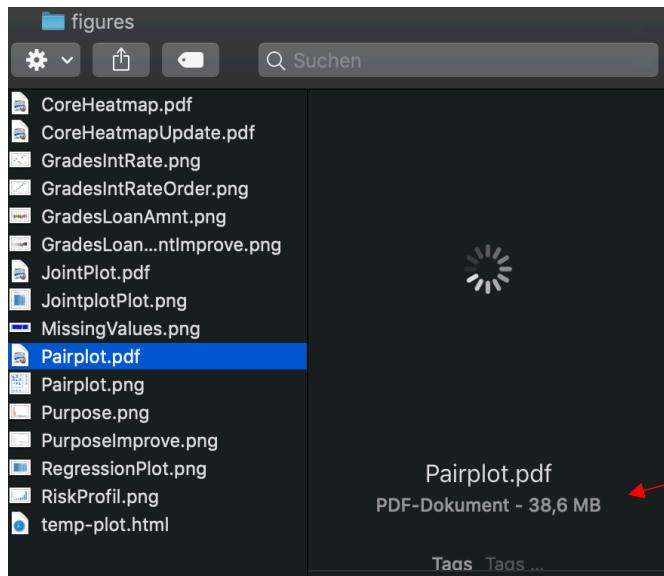
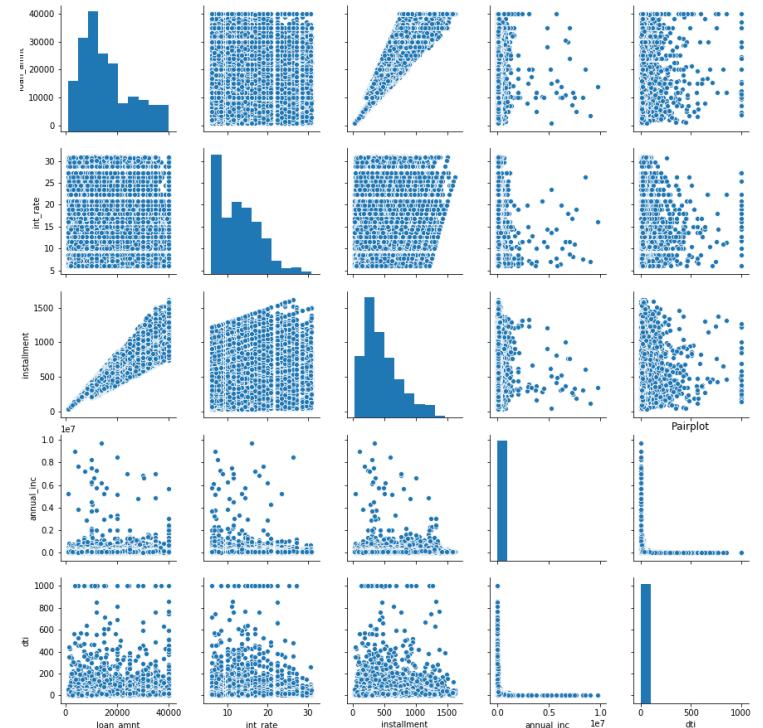
Get a Feeling of the Data – Pair Plot

- Plot pairwise relationships in a dataset
- Advantage:
 - Shows the pairwise relationship of data vectors from the Data Frame in the form of multiple scatter plots
 - Various analyses possible:
 - <https://seaborn.pydata.org/generated/seaborn.pairplot.html>
- Disadvantage:
 - Requires quantitative data (transformation)
 - Required, depending on data complexity, **computing time**



Solution: Pairplot

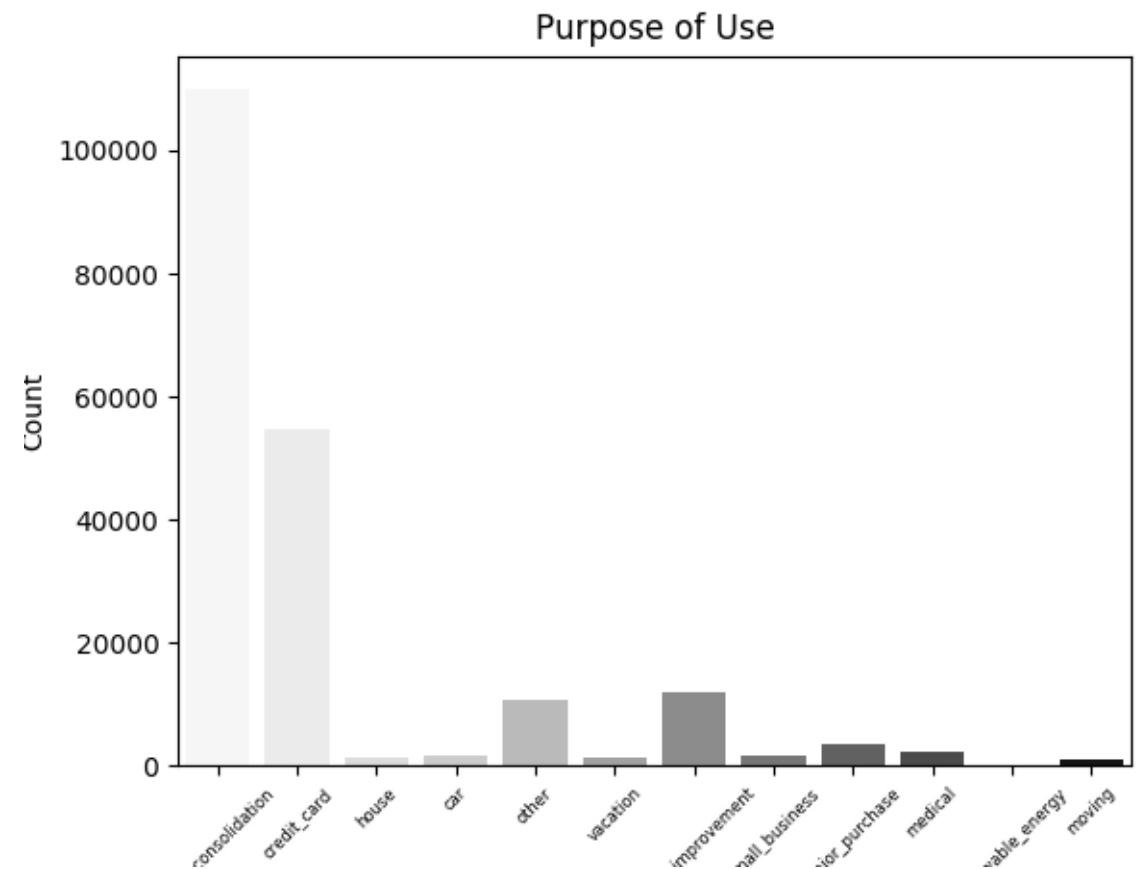
```
plt.figure()  
sns.pairplot(df[['loan_amnt', 'int_rate',  
'installment', 'annual_inc', 'dti']])  
plt.savefig('figures/Pairplot.png')  
plt.show()
```



Complex representation, this is also shown by the pdf memory requirement

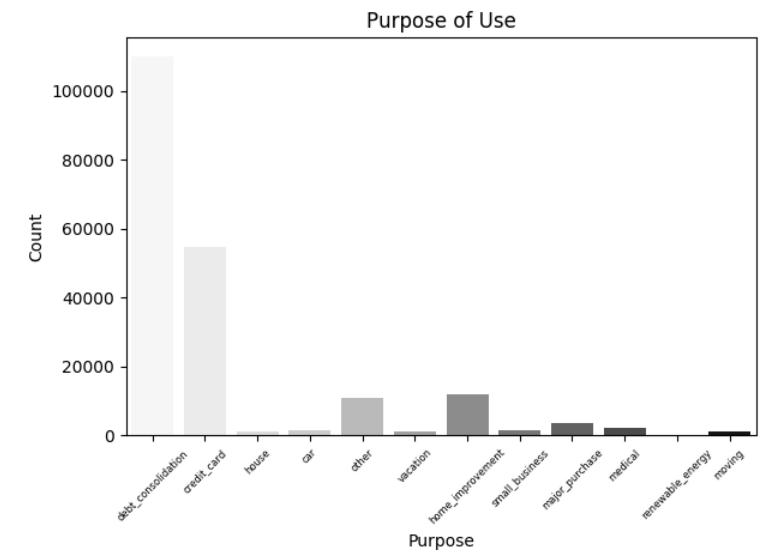
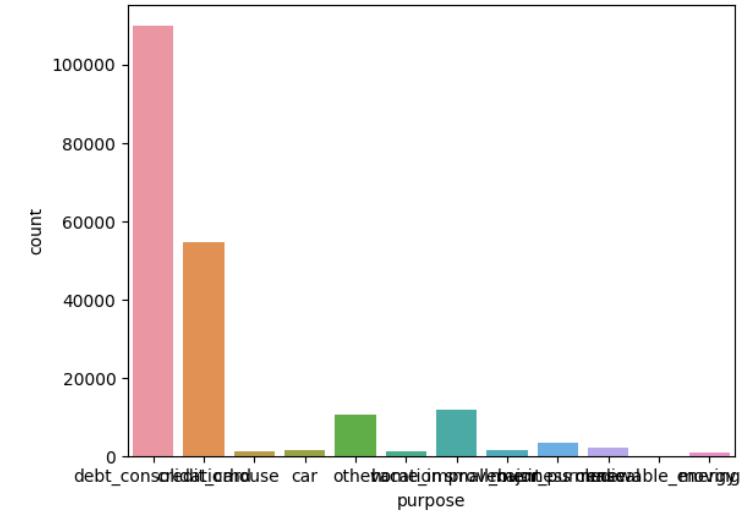
Ask an interesting question.

- Question: Purpose of Use?
- Why is this question relevant?
- Count Plot:
 - Show the counts of observations in each categorical bin using bars.



Solution: Upgrade Plot

```
ax = sns.countplot(x='purpose', data=df,
                    palette='Greys')
plt.xticks(rotation=45, fontsize=6)
plt.tight_layout()
plt.xlabel('Purpose')
plt.ylabel('Count')
plt.title('Purpose of Use')
plt.tight_layout()
plt.savefig('figures/PurposeImprov.png')
plt.show()
```



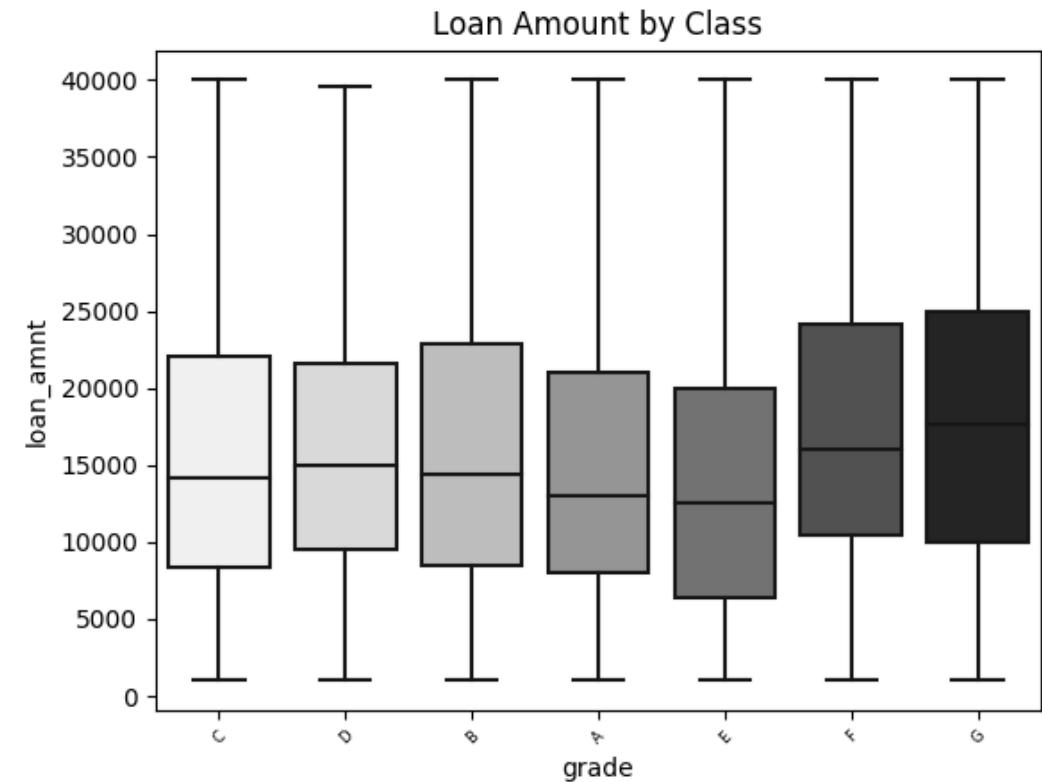
Risk Analysis

- A credit rating is an evaluation of the credit risk of a prospective debtor (an individual, a business, company or a government)
- The credit rating represents an evaluation of a credit rating agency of the qualitative and quantitative information for the prospective debtor
- Usually grouped in groups from A to F
 - A = very good
 - F = junk

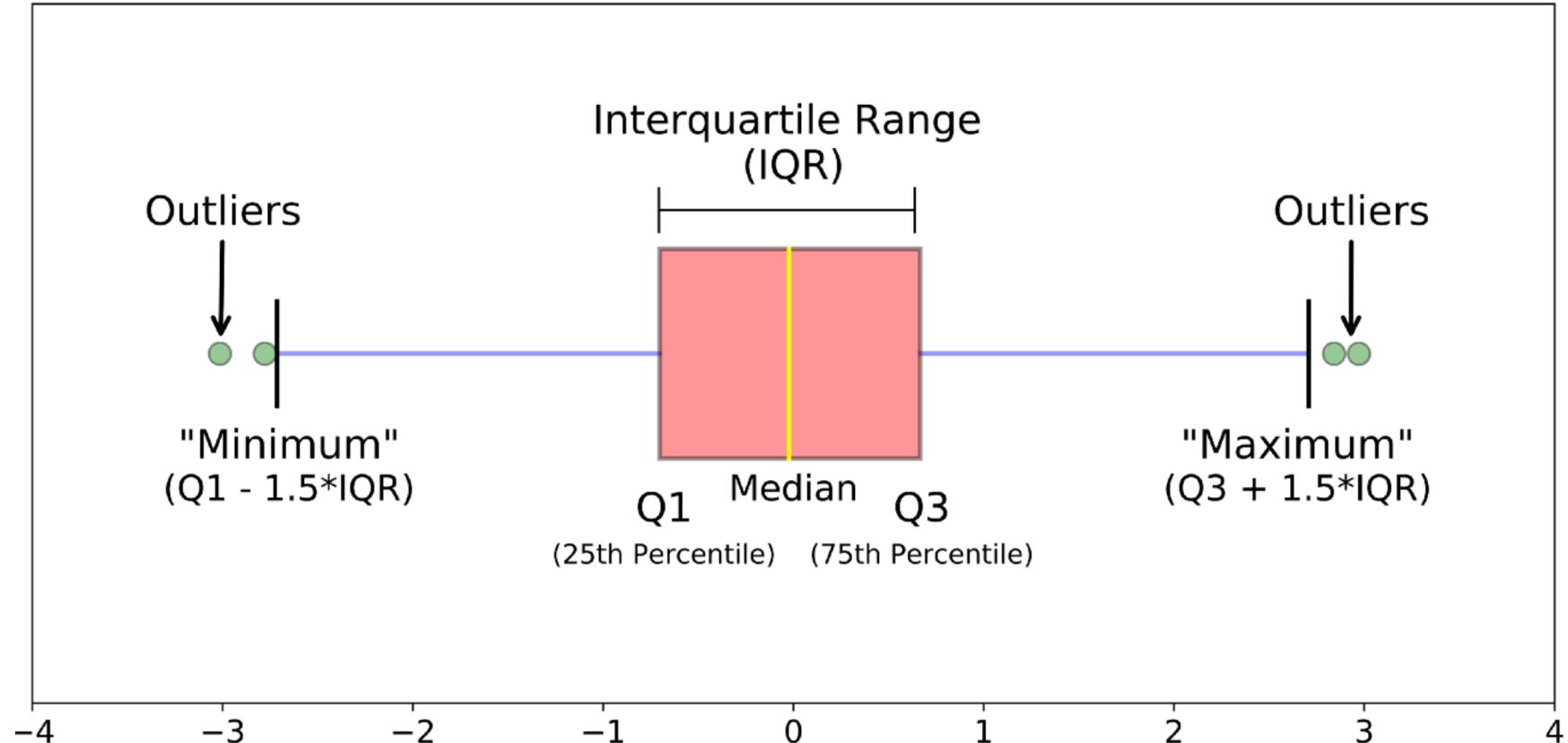


Ask an interesting question.

- Question: Loan amount by class?
- Why is this question relevant?
- Why could a boxplot be useful?



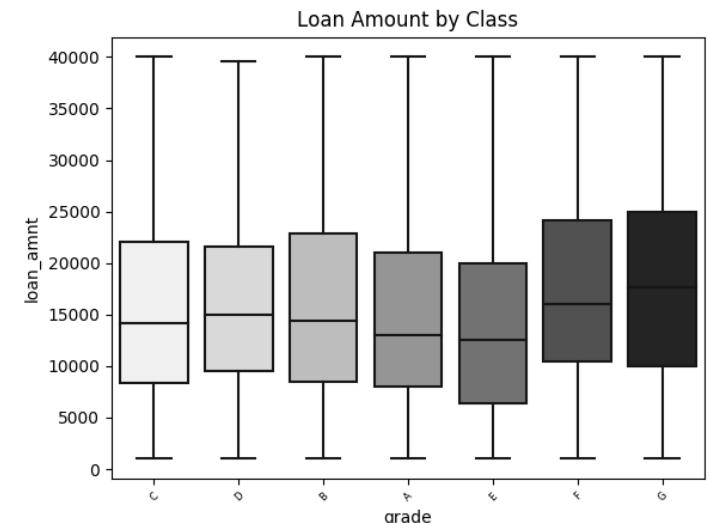
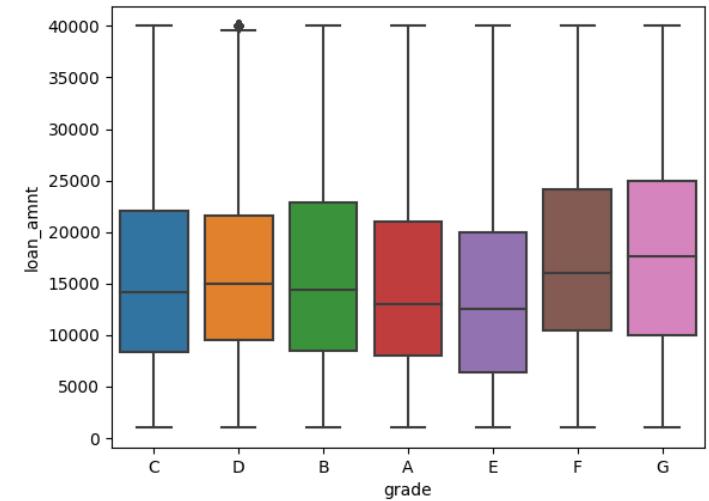
What is a Boxplot?



<https://towardsdatascience.com/understanding-boxplots-5e2df7bcbd51>

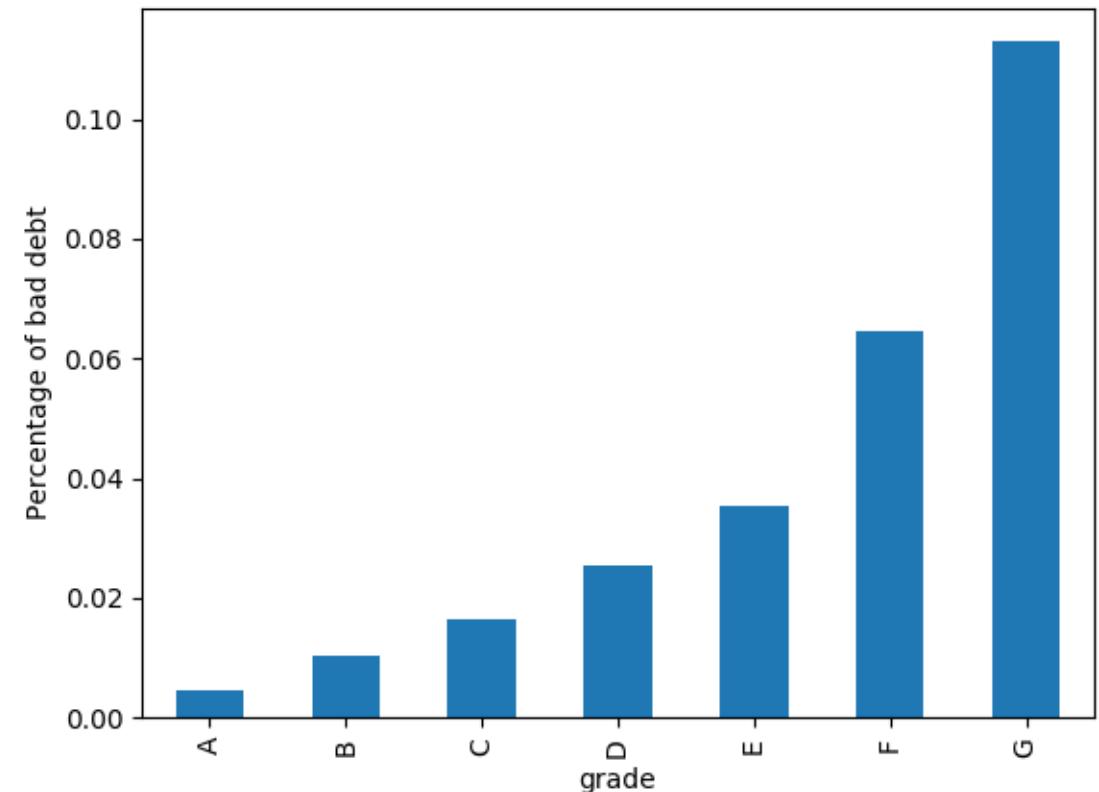
Upgrade Plot

```
plt.figure()
sns.boxplot(x='grade', y='loan_amnt',
data=df, showfliers=False,
palette='Greys')
plt.xticks(rotation=45, fontsize=6)
plt.title('Loan Amount by Class')
plt.savefig('figures/GradesLoanAmntImpro
ve.png')
plt.show()
```



Ask an interesting question.

- Question: Probability of loss?
- Why is this question relevant?



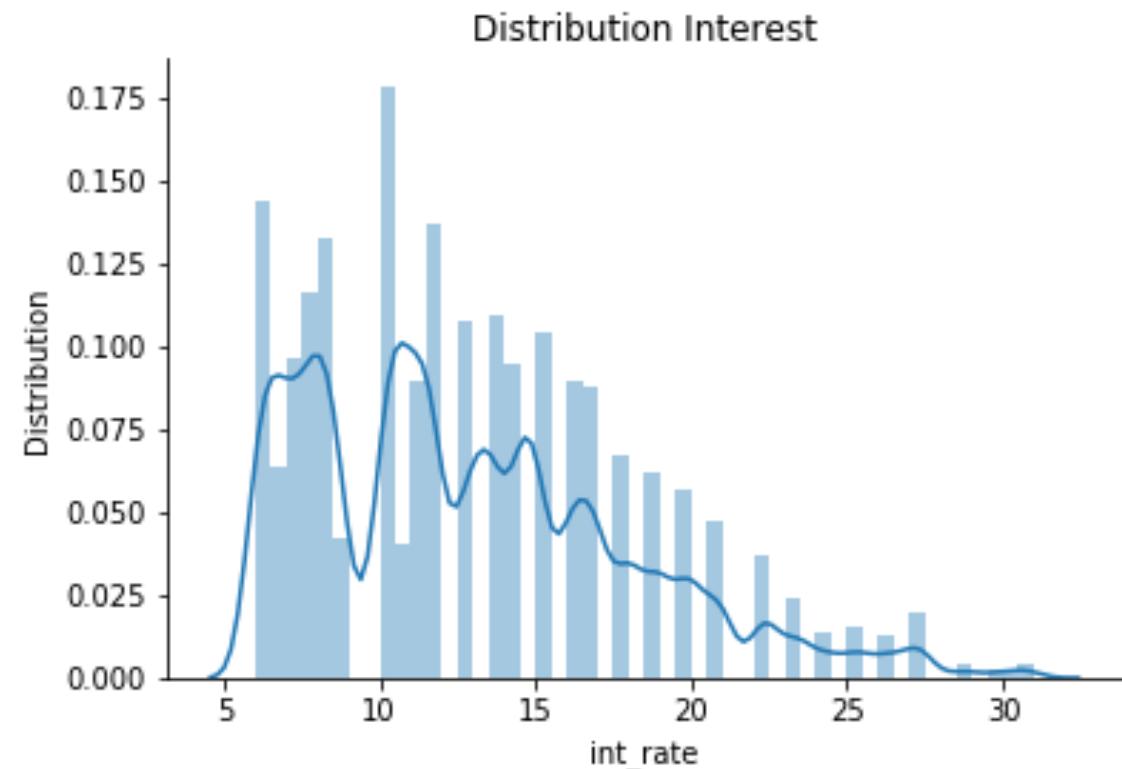
Return on Investment - Analysis

- Return on investment (ROI) is a ratio between net profit (over a period) and cost of investment (resulting from an investment of some resources at a point in time).



Distribution Interest

- Question: Distribution interest?
- Why is this question relevant?



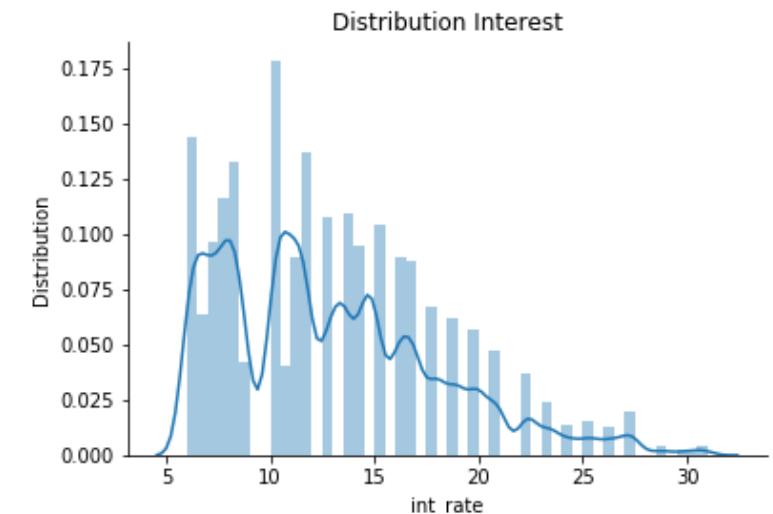
Upgrade Plot

```
fig, ax = plt.subplots()
sns.distplot(df['int_rate'])
plt.title('Distribution Interest')
plt.xlabel('int_rate')
plt.ylabel('Distribution')
ax.spines['right'].set_visible(False)
ax.spines['top'].set_visible(False)
plt.savefig('figures/Distplot.png')
plt.show()
```

fig : Figure

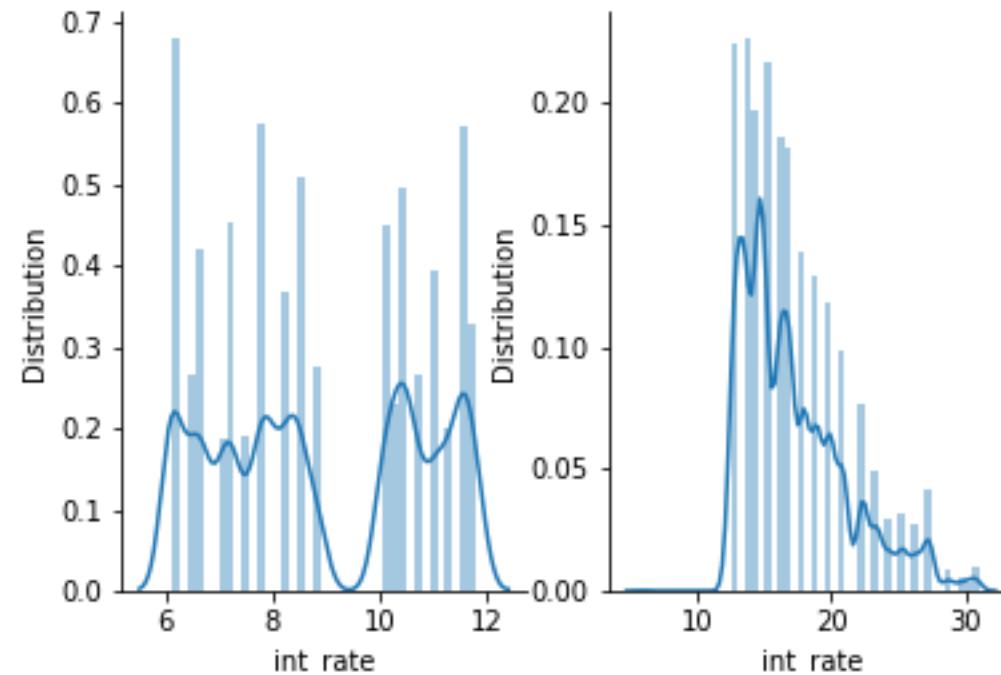
ax : axes.Axes object or array of Axes objects.

Create a figure and a set of subplots.



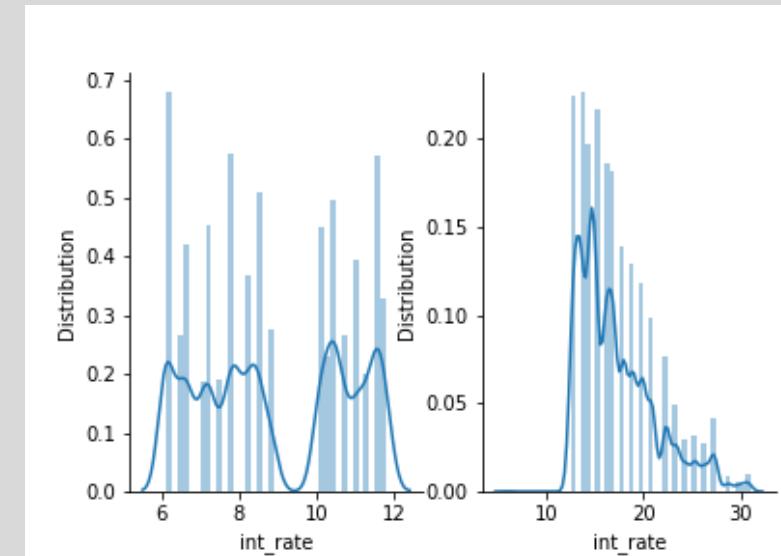
Distribution of Interest divided in two classes

- Lets take a deeper look!
- Using the cut method from DataFrame to create two classes of interest with the threshold of median.



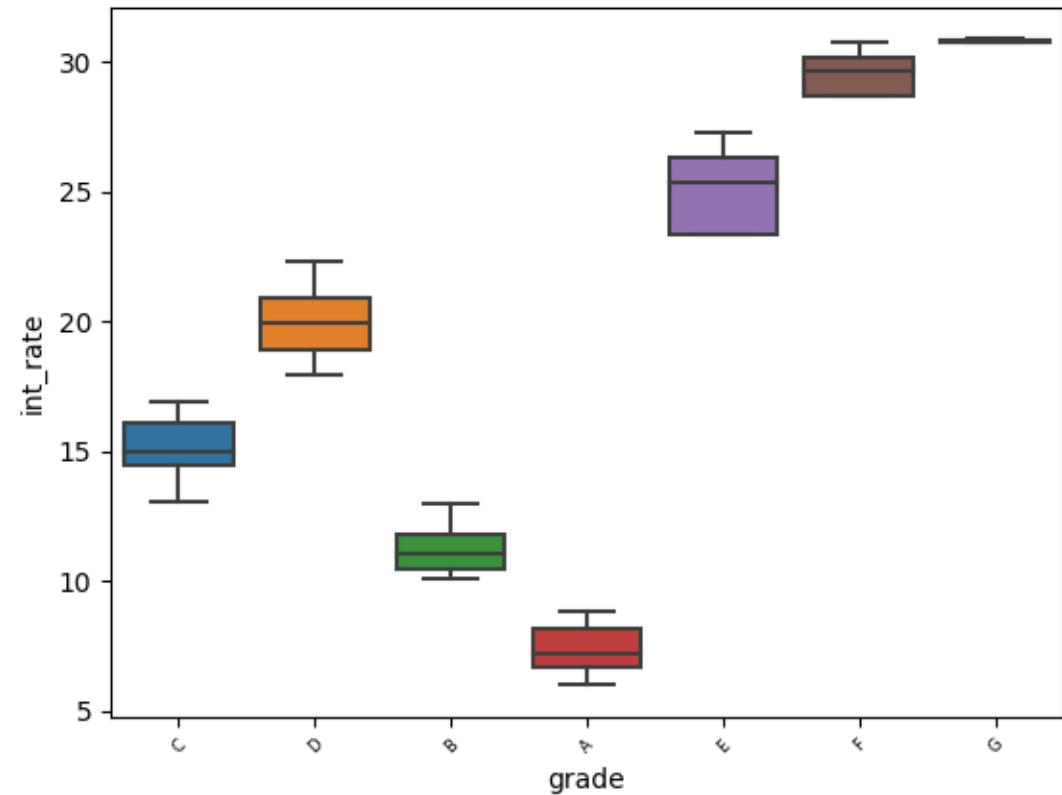
Distribution Interest divided in two classes

```
fig, ax = plt.subplots(1, 2)
sns.distplot( df['int_rate'][df['IntRateLH']=='lower'], ax=ax[0])
sns.distplot( df['int_rate'][df['IntRateLH']!='lower'], ax=ax[1])
ax[0].spines['right'].set_visible(False)
ax[0].spines['top'].set_visible(False)
ax[1].spines['right'].set_visible(False)
ax[1].spines['top'].set_visible(False)
ax[0].set_ylabel('Distribution')
ax[1].set_ylabel('Distribution')
plt.savefig('figures/SubPlotDistplot.png')
```



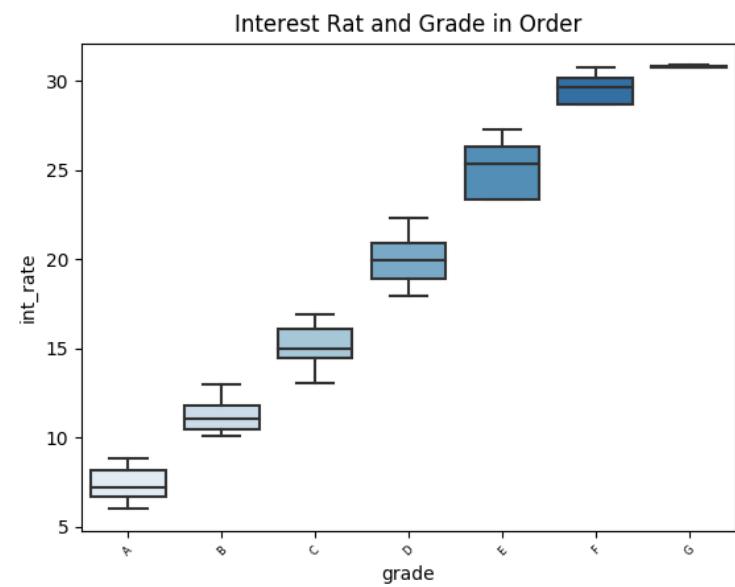
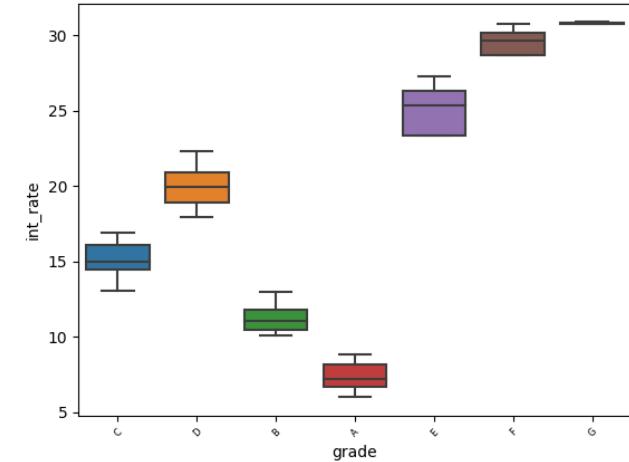
Ask an interesting question.

- Question: interest rate by class?
- Why is this question relevant?
- Why could be a boxplot useful?



Upgrade Plot

```
plt.figure()
sns.boxplot(x='grade', y='int_rate',
data=df, showfliers=False,
order=['A', 'B', 'C', 'D',
'E', 'F', 'G'])
plt.xticks(rotation=45, fontsize=6)
plt.title('Interest Rat and Grades in Order')
plt.savefig('figures/GradesIntRateOrder.
png')
plt.show()
```



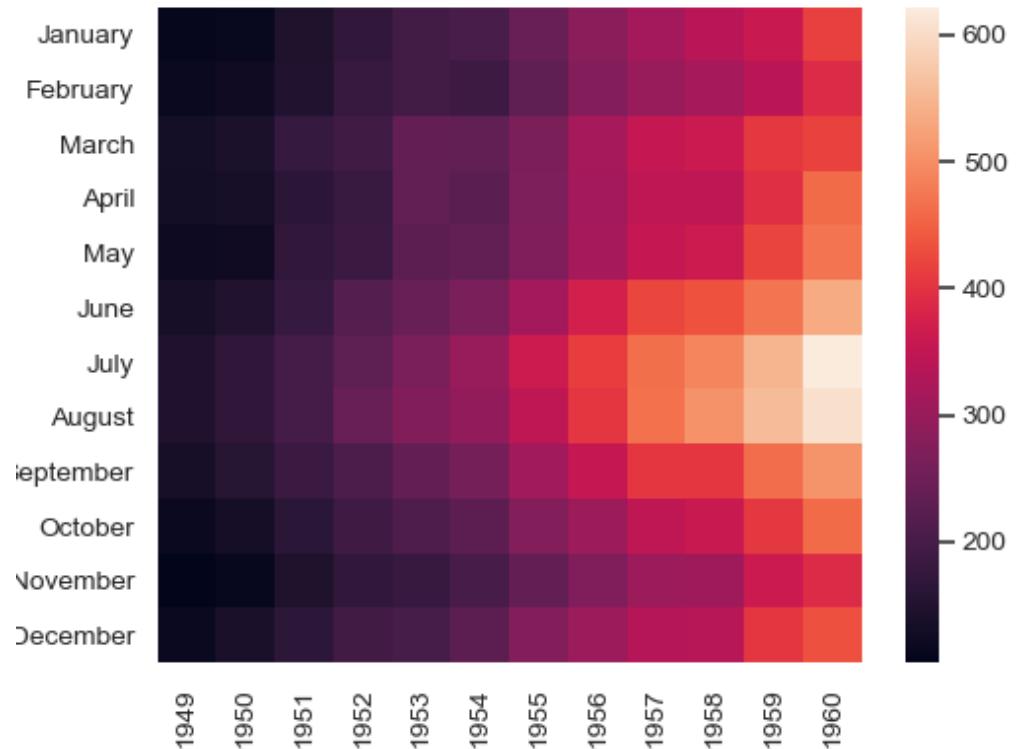
Other factors

7/29/2019



Show Heatmap

- A heatmap is a graphical representation of data where the individual values contained in a **matrix are represented as colours**.



Show Correlation with Heatmap

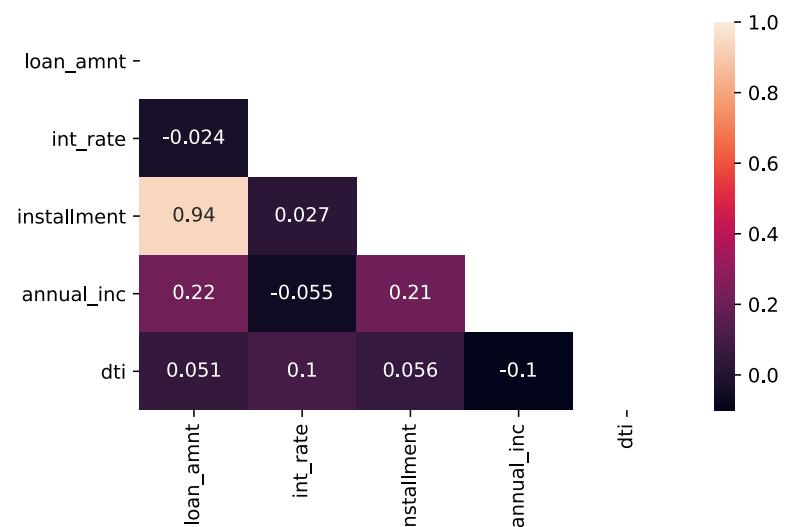
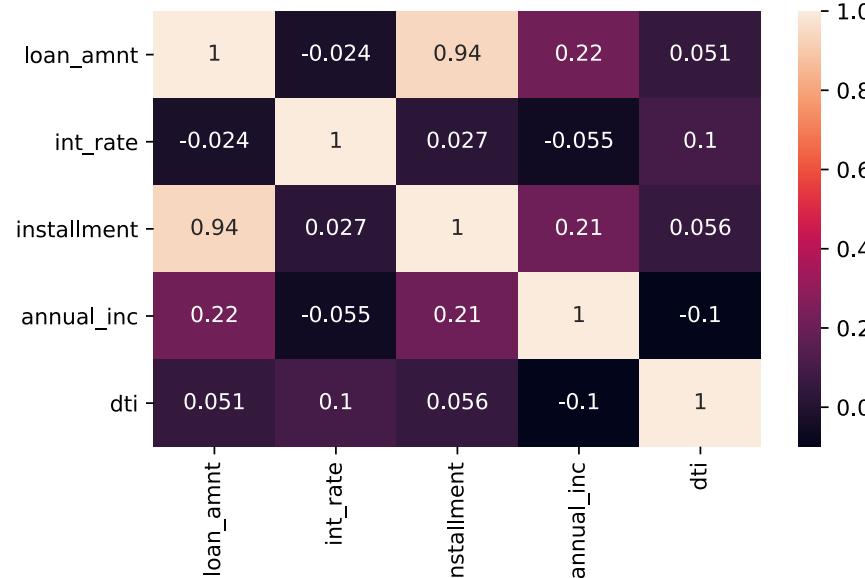
- **Correlation:**

a correlation coefficient is a numerical measure of some type of correlation, meaning a statistical relationship between two variables.

- **Calculation:**

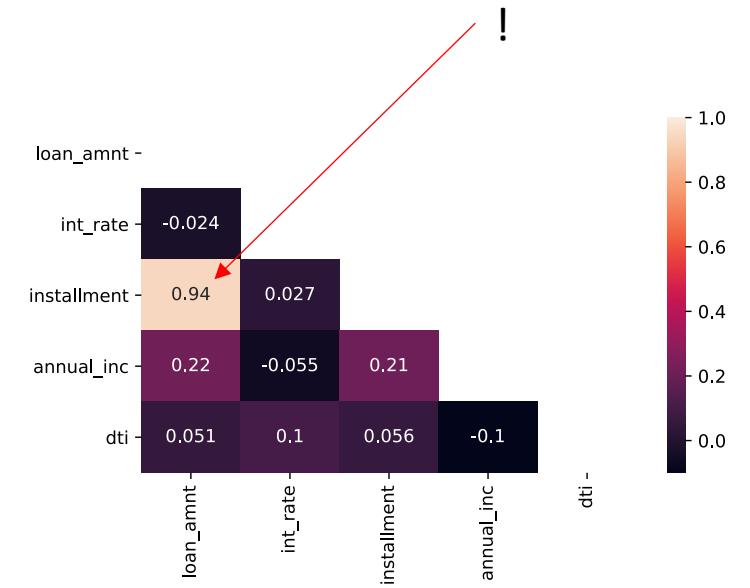
$$r_{xy} := \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \cdot \sum_{i=1}^n (y_i - \bar{y})^2}}$$

$r_{x,y}$ = has a value between +1 and -1

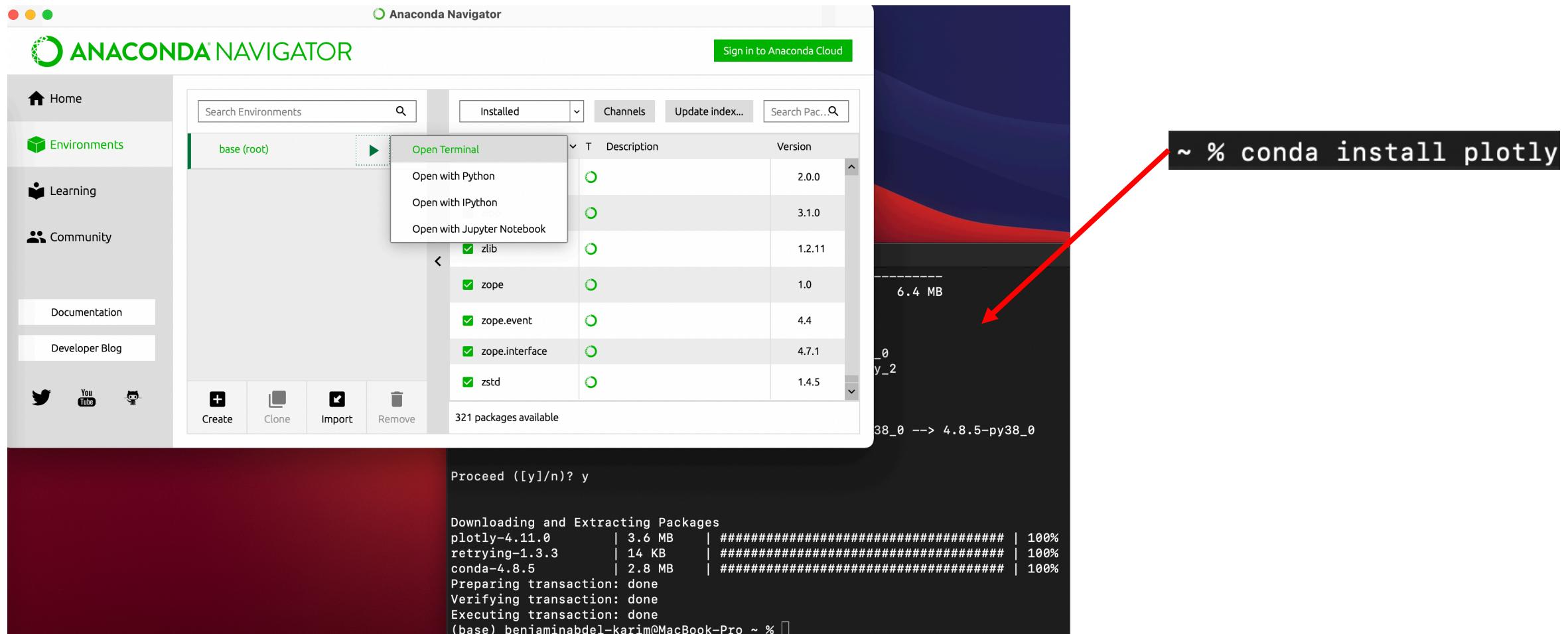


Upgrade Plot

```
plt.figure()
mask = np.zeros_like(corr,
dtype=np.bool)
mask[np.triu_indices_from(mask)] = True
sns.heatmap(corr,
            xticklabels=corr.columns,
            yticklabels=corr.columns,
            annot=True, mask=mask)
plt.savefig('CoreHeatmapUpdate.pdf')
plt.show()
```

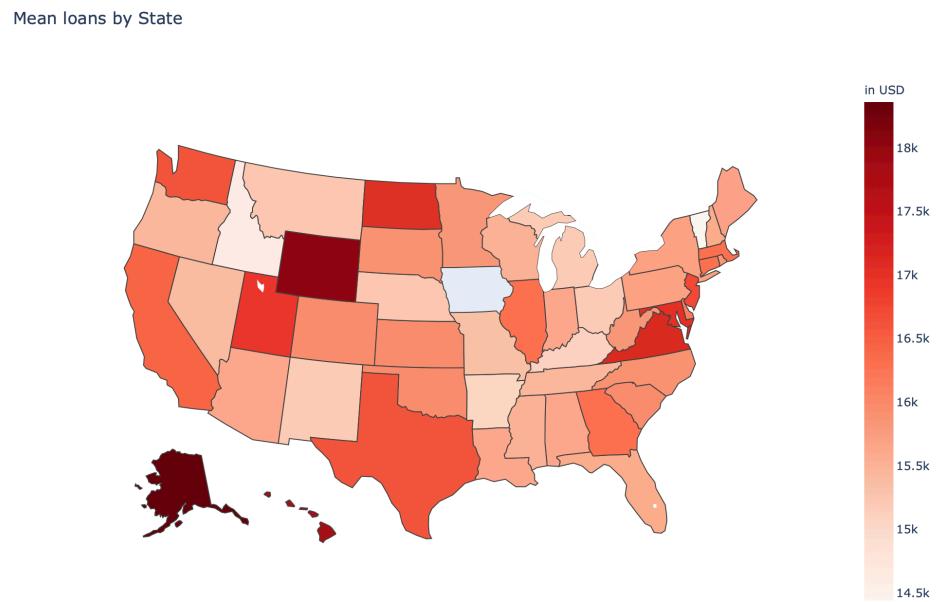


Preparing



Ask an interesting question.

- Question: What is the distribution of average credits based on zip codes?
 - Why is this question relevant?
 - Chloropleth (Choropleth map, in german: Flächenkartogramm): is a thematic map in which areas are shaded or patterned in proportion to the measurement of the statistical variable being displayed on the ma



Alternative: For Linux or mac user:
\$ conda install -c plotly plotly

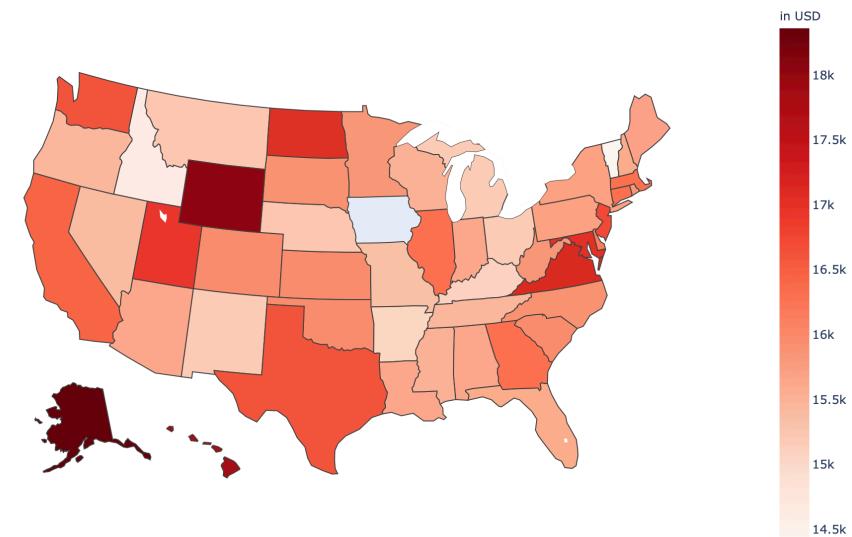
<https://anaconda.org/plotly/plotly>

Ask an interesting question.

- Question: What is the distribution of average credits based on zip codes?
- Why is this question relevant?

```
state_mean =  
pd.DataFrame(df.groupby('addr_state') ['loan_amnt'].mean())  
from plotly.offline import plot  
import plotly.graph_objects as go
```

Mean loans by State



Feedback



<https://limesurvey.emarkets.us/index.php?r=survey/index&sid=752265&lang=en>

Thank you!