Analysis of Collegiate Postgraduate Outcomes

Authors: Julia VanPutte (JHV42), Ellie Perlitz (EKP42), Tor Haugenes (TH455)

INFO 2950 Final Project, Fall 2022

Introduction

Determining what degree to obtain, what university to attend, and in which location to study in order to earn the highest paying salary is a question that many high school and college students consider. As Cornell students, we would also like to consider how receiving a degree at Cornell compares to recieving an equivalent degree at other schools around the United States, as it is important to know whether our degrees will pay us back in the future. In order to observe the salary outcomes for universities and colleges around the United States, we will use two different data sources to do an analysis of post graduate salary outcomes. We will specifically compare general national outcomes to Cornell salary outcomes and take degree, region, and college into consideration in this comparison. To conduct this analysis, We sourced a dataset from Kaggle.com containing data reported by the Wall Street Journal sources from Payscale that reports salary information by major of study, college attended, and region of school. This data was last updated 5 years ago, in 2017. To aid our comparison, we then scraped data from the Cornell post graduate outcomes dashboard to obtain median salary by major for Cornell undergraduates, looking specifically at 2016, 2017, and 2018 postgraduate outcomes to compare the same years.

In our data analysis to inform college students, we look to answer a number of questions:

- How do Cornell median postgraduate salaries by major compare to national postgraduate outcomes by major?
- Do STEM majors make more money than liberal arts majors?
- How does college type affect starting salary and mid career salary?
- How do salaries increase over time?
- What is the best combination of major and college that will yield the highest paying salary, both mid career and starting salary?
- Which region of attending college in the US allows graduates to have the highest starting salary?
- Which majors pay back the most in starting salaries at Cornell?

Based on our analysis, we conclude through statistical A/B testing that the median postgraduate salaries at Cornell are higher than national postgraduate salaries, with computer science being the highest paid starting starting salary from Cornell. Further, logistic regression showed that STEM majors tend to make more money that liberal arts (non-STEM) majors, with majoring in a STEM field having a more drastic impact on starting salaries than mid-career salaries. Therefore, when looking into what to major in at college, there is a high chance that a STEM degree will allow you to earn a higher salary, especially when you are entering the workforce for the first time. Additionally, when looking at types of schools, we were also able to conclude that attending an engineering or Ivy League school will result in a higher paying salary. This analysis was done through Naive Bayes classification. Further, as predicted by linear regression, salaries increase over time, and having a higher starting salary can often result in having a higher mid-career salary

as well. Looking further into the specifics, obtaining an engineering degree at a school in the Northeast or in California will result in the highest paying starting and mid-career salary. Physician's Assistant is the highest paying starting salary nationally, Chemical engineering is the highest paying mid-career salary in national outcomes, California Institute of Technology has the highest paying starting salary, and Dartmouth has the highest paying mid career salary. Thus, this combination will likely yield among the highest salaries nationally.

Overall, our project seeks to understand how different colleges, majors, and regions of study impact starting salary earnings and mid-career salary earnings. By comparing this with Cornell data, we find that studying at Cornell generally yields a higher paying job than other institutions. Further, engineering and Ivy League schools also result in higher pay, and STEM majors reflect higher compensation as well. However, this is not to say that other schools and other majors do not yield high paying jobs, as much of our comparisons were done based on median salaries. For further study, we would want to look into individual student datapoints corresponding to salaries. However, this data is not available due to privacy concerns. Therefore, we have presented a broad overview of how your college, major, and region will impact your future earnings.

Data Collection and Cleaning

To begin our analysis, we describe how we sourced and cleaned our data to ready it for analysis. We downloaded the national college dataset from kaggle.com (link : https://www.kaggle.com/datasets/wsj/college-salaries?select=degrees-that-pay-back.csv). First, we converted all values to strings in order to remove all of the '$'s from the data. Then, we converted all numeric values back to integers for salary statistics. We also removed all NaNs from the data and replaced these with 0 so that our type conversion would not fail. We will handle the 0's accordingly later during analysis. We also added percent change columns to data frames that did not originally include this through a simple calculation. We then exported these cleaned dataframes as csv files, so that the cleaned csv files could be imported into the final notebook. The cleaned csv files are uploaded to our github repository for reference.

We scraped the Cornell dataset manually from Cornell's postgraduate outcomes dashboard (link : https://ccs.career.cornell.edu/dash/dashboard_activity). This was done manually since we did not have access to the full csv data and had to filter by major on Cornell's dashboard in order to get statistics for salaries per major. To pick each major, we filtered by years 2016, 2017, and 2018, Bachelor's level degree, and found majors at Cornell that generally corresponded to the national majors from our dataset. Because Cornell has somewhat unique majors, we did our best to match up majors nationally with majors at Cornell, and included the matched majors in our dataset. We just imported salary statistics and not educational outcomes, because we are only focusing on salary statistics currently. Once the Cornell data was downloaded as a csv file, we then converted the types to ensure accurate analysis. The dataset we scraped can also be referenced at our repository.

Data Description

We detail a description of our dataset here to inform the reader in detail about the Motivation, Composition, Collection, Preproccessing, Cleaning, Labeling, and Uses of our datasets. The questions answered in this report correspond to the "Datasheets for Datasets" paper referenced for complete data description.

Motivation

For what purpose was the dataset created? Was there a specific task in mind? Was there a specific gap that needed to be filled? Please provide a description.

- The salary by type of college dataset was created in order to observe how starting salary is affected by types of schools people attend such as party schools, ivy league school, liberal arts colleges, and state schools.
- The salary by region dataset was created to observe how attending school in a certain region could affect salaries post graduation and mid career.
- The salary by major dataset was created to assess which degrees pay back the most post graduation.
- The Cornell postgraduate outcomes dashboard was created so that people can filter by year, major, school, and degree level to see what the salary trends are for Cornell students post graduation and where students typically end up. The dataset that we extracted from this dashboard was created so that we could look at which majors specifically at Cornell pay the highest/lowest salaries.

Who created the dataset (e.g., which team, research group) and on behalf of which entity (e.g., company, institution, organization)?

- The salary by region, salary by college type, and salary by major datasets were all created by the Wall Street Journal, from data collected by Payscale Inc on individuals with only bachelor's degrees
- This data was collected through a year-long survey of 1.2 million people with only a bachelor's degree by PayScale Inc
- All data was obtained from the Wall Street Journal based on data from Payscale, Inc.
- The Cornell outcomes dataset was manually created by Julia VanPutte and Ellie Perlitz by filtering on the Cornell Outcomes Dashboard to get starting salary statistics for each major at Cornell that corresponded to majors in the WSJ salary by major dataset in order to conduct a comparison analysis in the future

Who funded the creation of the dataset? If there is an associated grant, please provide the name of the grantor and the grant name and number.

- The dataset creation was funded by the Wall Street Journal
- The Cornell Postgraduate data dashboard was funded by the Cornell Career Services
- Ellie Perlitz and Julia VanPutte were not compensated for their manual dataset construction.

Composition

What do the instances that comprise the dataset represent (e.g., documents, photos, people, countries)? Are there multiple types of instances (e.g., movies, users, and ratings; people and interactions be- tween them; nodes and edges)? Please provide a description.

- In the Salary by Region dataset, the instances are colleges, with descriptions of the region of the college and information about starting and mid career median salaries
- In the Salary by College dataset, the instances are colleges, with descriptions of the type of the college and information about starting salary and mid career median salaries
- In the Degrees that pay back dataset, each instance corresponds to a major in college, with data about mid career median salaries and starting median salaries per major.

- In the Manually created Cornell majors dataset, each instance is a Cornell major that maps to a major in the degrees that pay pack dataset, with statistics about Cornell starting salaries for each major.

How many instances are there in total (of each type, if appropriate)?

- Salary by Region = 320
- Salary by College = 269
- Note: the salaries by college has less instances than the salary by region, but both of these correspond to instances of colleges. We will search for duplicate instances. The salary by region dataset is more complete.
- Degrees that Pay Back = 50

index	Undergraduate Major
0	Accounting
1	Aerospace Engineering
2	Agriculture
3	Anthropology
4	Architecture
5	Art History
6	Biology
7	Business Management
8	Chemical Engineering
9	Chemistry
10	Civil Engineering
11	Communications
12	Computer Engineering
13	Computer Science
14	Construction
15	Criminal Justice
16	Drama
17	Economics
18	Education
19	Electrical Engineering
20	English
21	Film
22	Finance
23	Forestry
24	Geography
25	Geology
26	Graphic Design

index	Undergraduate Major
27	Health Care Administration
28	History
29	Hospitality & Tourism
30	Industrial Engineering
31	Information Technology (IT)
32	Interior Design
33	International Relations
34	Journalism
35	Management Information Systems (MIS)
36	Marketing
37	Math
38	Mechanical Engineering
39	Music
40	Nursing
41	Nutrition
42	Philosophy
43	Physician Assistant
44	Physics
45	Political Science
46	Psychology
47	Religion
48	Sociology
49	Spanish

- Cornell Majors = 40
- Note: there were 10 majors in the national dataset (Degrees that pay back) that do not map to Cornell majors as Cornell does not offer these majors. We omitted these majors from our comparison with Cornell data.

Does the dataset contain all possible instances or is it a sample (not necessarily random) of instances from a larger set? If the dataset is a sample, then what is the larger set? Is the sample representa- tive of the larger set (e.g., geographic coverage)? If so, please describe how this representativeness was validated/verified. If it is not representative of the larger set, please describe why not (e.g., to cover a more diverse range of instances, because instances were withheld or unavailable).

- The datasets are samples.
- The Cornell dataset is a subset of majors, where the larger set is all majors at Cornell
- The Salaries by Region and Salaries by College is a subset of major colleges, where the larger set is every postsecondary institution in the United States
- The Degrees that pay back is a subset of all college majors across the nation. The WSJ just chose the most common majors.

What data does each instance consist of? “Raw” data (e.g., unprocessed text or images) or features? In either case, please provide a description.

- Each instance consists of features (represents measurable and analyziable data)

These features are detailed below:

Degrees that pay back, each instance consists of:

- Undergraduate Major (String)
- Starting Median Salary
  - Median salary for each major (Float)
- Percent change from Starting to Mid-Career Salary (Float)
  - =100*(Starting salary - Mid Career Salary)/Starting Salary
- Mid-Career Median Salary (Float)
- Mid-Career 10th Percentile Salary (Float)
- Mid-Career 25th Percentile Salary (Float)
- Mid-Career 75th Percentile Salary (Float)
- Mid-Career 90th Percentile Salary (Float)
- Salaries by region, each instance consists of
- School Name (College Name) (String)

Region (California, Western, Southern, Midwestern, Northeastern) (String)

- Starting Median Salary
  - Median salary for each university (Float)
- Percent change from Starting to Mid-Career Salary (Float)
  - =100*(Starting salary - Mid Career Salary)/Starting Salary
- Mid-Career Median Salary (Float)
- Mid-Career 10th Percentile Salary (Float)
- Mid-Career 25th Percentile Salary (Float)
- Mid-Career 75th Percentile Salary (Float)
- Mid-Career 90th Percentile Salary (Float)

Salary by type of college, each instance consists of

- School Name (College Name) (String)
- School Type (Engineering, Party, Liberal Arts, State, Ivy League) (String)
- Starting Median Salary
  - Median salary for each university (Float)
- Percent change from Starting to Mid-Career Salary (Float)
  - =100*(Starting salary - Mid Career Salary)/Starting Salary
- Mid-Career Median Salary (Float)
- Mid-Career 10th Percentile Salary (Float)
- Mid-Career 25th Percentile Salary (Float)
- Mid-Career 75th Percentile Salary (Float)
- Mid-Career 90th Percentile Salary (Float)

Cornell Starting salary by major dataset, each instance consists of

- Cornell Major
  - Name of Major(s) at Cornell corresponding to national major (String)
- National Data Comparison Major
  - National major from WSJ dataset that we matched with Cornell Major (String)
- Mean Salary
  - Mean starting salary for Cornell Grads in this major (Float)
- Median Salary
  - Median Starting Salary (Float)
- Mode
  - Most frequent starting salary for starting salary Cornell graduate in each major (Float)
- Lower Range
  - Lowest starting salary from Cornell of each major (Float)
- Upper Range
  - Highest starting salary from Cornell of each major (Float)
- STD
  - Standard deviation of starting salary (Float)
- Number of Responses (Integer)
  - Number of cornell students included in the outcomes survey
- Dates
  - Dates considered from the Cornell outcomes dashboard to compute salary statistics (Datetime range)

Is there a label or target associated with each instance? If so, please provide a description.

- Target- the starting and Mid career salary could be targets associated with each instance
- Labels: None

Is any information missing from individual instances? If so, please provide a description, explaining why this information is missing (e.g., because it was unavailable). This does not include intentionally removed information, but might include, e.g., redacted text.

- Yes, there are some missing values in the Salaries by Region and the Salaries by type of college dataset. In these datasets, we have complete values for all starting salary median and mid career salary median, but sometimes the 10% and 90% quantiles are not reported. This information was unavailable as provided to us from the data online. We could try to calculate quantiles but without data about the standard deviations of each salary from the survey, this would be difficult.

Are relationships between individual instances made explicit (e.g., users' movie ratings, social network links)? If so, please describe how these relationships are made explicit.

- There are no relationships made explicit between individual instances.

Are there recommended data splits (e.g., training, development/validation, testing)? If so, please provide a description of these splits, explaining the rationale behind them.

- No, there were no recommended data splits

Are there any errors, sources of noise, or redundancies in the dataset? If so, please provide a description.

- No, there are no sources or noise or redundancies in each dataset.

Is the dataset self-contained, or does it link to or otherwise rely on external resources (e.g., websites, tweets, other datasets)?

- The dataset is self-contained because there are no links that lead to external sources.

Does the dataset contain data that might be considered confidential (e.g., data that is protected by legal privilege or by doctor– patient confidentiality, data that includes the content of individuals' non-public communications)? If so, please provide a description.

- No, there is no confidential data in the data set.

Does the dataset contain data that, if viewed directly, might be offensive, insulting, threatening, or might otherwise cause anxiety? If so, please describe why.

- The data is not offensive, insulting, or threatening, but it might cause anxiety if someone realizes that their major, college, region, etc might cause them to have a low starting and/or mid year salary.

Collection

How was the data associated with each instance acquired? Was the data directly observable (e.g., raw text, movie ratings), reported by subjects (e.g., survey responses), or indirectly inferred/derived from other data (e.g., part-of-speech tags, model-based guesses for age or language)? If the data was reported by subjects or indirectly inferred/derived from other data, was the data validated/verified? If so, please describe how.

- The WSJ data was acquired from Payscale Inc, and Payscale acquired this data through a year-long survey of 1.2 million people and took median and quantile values for salaries from this survey, grouping by college, region, and major
- The Cornell Data was acquired from the Cornell outcomes dashboard of reported starting salary statistics from Cornell graduates in each major.

What mechanisms or procedures were used to collect the data (e.g., hardware apparatuses or sensors, manual human curation, software programs, software APIs)? How were these mechanisms or procedures validated?

- Data was collected from the Cornell postgraduate outcomes dashboard through manual human curation. This was validated through double-checking over the data once we had collected it in a google sheet. The original data that generated the dashboard was collected through surveys of recent graduates.
- Data was collected for the national college salaries public dataset through the use of surveys of recent graduates from college of a specific major.

If the dataset is a sample from a larger set, what was the sampling strategy (e.g., deterministic, probabilistic with specific sampling probabilities)?

- The WSJ data was sampled using most common majors and colleges from samples of all majors and all colleges. We do not know why certain colleges and majors were chosen
- The Cornell dataset was sampled in order to have correspondence between majors at Cornell and majors in the national WSJ dataset. We only chose majors that corresponded to majors in the WSJ dataset.

Who was involved in the data collection process (e.g., students, crowdworkers, contractors) and how were they compensated (e.g., how much were crowdworkers paid)?

- Information Not Available for WSJ, Cornell students provide survey answers that guide the Cornell data.

Over what timeframe was the data collected? Does this timeframe match the creation timeframe of the data associated with the instances (e.g., recent crawl of old news articles)? If not, please describe the timeframe in which the data associated with the instances was created.

- The Cornell postgraduate outcomes dashboard data comes primarily from 2016-2018. There are some instances where the major was very rare and there were limited instances so we had to use all data including up to and including the current year.
- The national college salaries public dataset is comprised of data that was last updated five years ago (approximately October 2017). There is no information with regards to what year the dataset takes into account as a starting point.

Were any ethical review processes conducted (e.g., by an institutional review board)? If so, please provide a description of these review processes, including the outcomes, as well as a link or other access point to any supporting documentation.

- Not that we know of.

Preprocessing/cleaning/labeling

Was any preprocessing/cleaning/labeling of the data done (e.g., discretization or bucketing, tokenization, part-of-speech tagging, SIFT feature extraction, removal of instances, processing of miss- ing values)? If so, please provide a description.

- We downloaded the national dataset from kaggle.com (link : https://www.kaggle.com/datasets/wsj/college-salaries?select=degrees-that-pay-back.csv). First, we converted all values to strings in order to remove all of the '$' from the data. Then, we converted all numeric values back to integers for salary statistics. We also removed all Nans from the data and replaced these with 0 so that our type conversion would not fail. We will handle the 0's accordingly later during analysis. We also added percent change columns to data frames that did not originally include this through a simple calculation.
- We scraped the Cornell dataset manually from Cornell's postgraduate outcomes dashboard (link : https://ccs.career.cornell.edu/dash/dashboard_activity). This was done manually since we did not have access to the full csv data and had to filter by major on Cornell's dashboard in order to get statistics for salaries per major. To pick each major, we filtered by years 2016,2017, and 2018, Bachelor's level degree, and found majors at Cornell that generally corresponded to the national majors from our dataset. Because Cornell has somewhat unique majors, we did our best to match up majors nationally with majors at Cornell, and included the matched majors in our dataset. We just imported salary statistics and not educational outcomes, because we are only focusing on salary statistics currently. If more data is needed, we can scrape more data from the Cornell outcomes dashboard and analyze that in the future. Once the Cornell data was downloaded as a csv file, we then converted the range of dates to DateTime objects. We also had to convert the majors to strings.

Was the "raw" data saved in addition to the preprocessed/cleaned/labeled data (e.g., to support unanticipated future uses)? If so, please provide a link or other access point to the "raw" data.

- The raw data was saved.
- We have the raw csvs uploaded to our project repository. https://github.com/juliavanputte7/2950 and to our google drive folder

Is the software that was used to preprocess/clean/label the data available? If so, please provide a link or other access point.

- Python pandas package was used to clean the data

Uses

Has the dataset been used for any tasks already? If so, please provide a description.

- The Cornell postgraduate outcomes dashboard (to our knowledge) has not been used for any tasks other than display data on the outcomes of students from specific majors/schools to illustrate what industries they work in and what type of salaries they make initially.
- The national college salaries dataset has not been used for any tasks to our knowledge recently and we can not find any real-life use cases on the Kaggle posting of this data set. Other than having been used to display how salary is impacted by “Type of school”, “Region”, and “Major”

Is there a repository that links to any or all papers or systems that use the dataset? If so, please provide a link or other access point.

- To our knowledge, there is no repository available that links all systems that use the datasets.

What (other) tasks could the dataset be used for?

- The Cornell postgraduate outcomes dashboard is primarily used for helping students see what kinds of careers specific majors lead to. This sole purpose is why this tool exists. When trying to think of additional tasks that this dataset could be used for, the idea of using the data to determine what degree(s) hold the most (monetary) potential value comes to mind.
- The national college salaries dataset has similar uses to the dashboard and is mainly used for providing information on what majors/degrees hold the most potential value when examining from a purely financial standpoint. I cannot think of any other uses of this dataset, as it is pretty specific to that specific task.

Is there anything about the composition of the dataset or the way it was collected and preprocessed/cleaned/labeled that might impact future uses? For example, is there anything that a dataset consumer might need to know to avoid uses that could result in unfair treatment of individuals or groups (e.g., stereotyping, quality of service issues) or other risks or harms (e.g., legal risks, financial harms)? If so, please provide a description. Is there anything a dataset consumer could do to mitigate these risks or harms?

- We don’t believe that there was anything about the way it was collected and composed that impacts future uses. There is no way the dataset could lead to the unfair treatment of individuals because it does not deal with individuals. The only concern here would be making sure that the data is current so that it’s findings are accurate descriptions of the current situation. If the salary/major data is inaccurate, it could be harmful and misleading for individuals seeking to pursue a career in a major that is inaccurately represented by the data.

Are there tasks for which the dataset should not be used? If so, please provide a description.

- I cannot think of any tasks for which the dataset should not be used. The only case I can think of would be using this data to determine which majors get paid the least amount in the hopes of making people of that major feel bad (there are moral/ethical concerns with doing this that I hope are obvious).

Kaggle WSJ Dataset:

All data was obtained from the Wall Street Journal based on data from Payscale, Inc

This data was last updated in 2017

Salary Increase by Type of College : 269 data points

- All NaNs replaced with 0s, all '$' and ',' removed Salaries by Region : 320 records
- All NaNs replaced with 0s, all '$' and ',' removed Salary Increase by Major : 50 records
- All NaNs replaced with 0s, all '$' and ',' removed

Cornell Dataset

Data scraped from https://ccs.career.cornell.edu/dash/reset corresponding to Cornell postgraduate outcomes that correspond to majors listed in the salaries by major dataset.

Preregistration of Analyses Statement

Hypothesis 1:

Cornell Postgraduate median salaries are greater than national postgraduate median salaries for more than 95% of majors.

This will help us to understand how the degrees at Cornell compare to those nationwide and if obtaining these degrees are worth the money at Cornell vs. other schools.

Hypothesis 2:

For national data, STEM majors make more money both in starting salaries and over time than liberal arts majors.

If STEM majors make more money to a large extent than liberal arts majors, it is important to take this into consideration when choosing a field at college. However, if our hypothesis is incorrect and this is not a valid conclusion, majoring in STEM may not be worth it.

Hypothesis 3:

Ivy league graduates have greater earnings median salaries over time compared to other types of schools.

When looking at low acceptance rates of ivy leagues, it is important to take into consideration whether the pressure of getting into an ivy league school is worth it in the long run. Therefore, if our hypthesis is correct, than trying to get into an ivy league school could be worth it for obtaining a higher salary in the future. If not, it is not worth the stress and pressure of getting into and attending an ivy league school.

Hypothesis 4:

Salaries increase over time, and higher starting salaries can determine higher mid-career salaries.

It is often a question as to whether a starting salary will determine the progression of a career and will allow for higher mid-career salaries down the road. If this hyothesis is true, knowing that salaries will increase over time will alleviate the pressure of getting a high starting salary. However, seeing what kind of impact starting salaries will have will determine how important it is to have a higher starting salary for future success.

Further Analysis:

We will present the best combination of major and college that will yield the highest paying salary, both mid career and starting salary.

This will allow to consideration of the best combination of school, region, and degree that will pay back the most post college/university.

Data Analysis

In [28]:
```
#import libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import datetime as datetime

import seaborn

from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer
from sklearn.preprocessing import normalize

from sklearn.cluster import KMeans
from sklearn.naive_bayes import GaussianNB
from sklearn.linear_model import LinearRegression
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn import metrics
from scipy.stats import ttest_ind
from scipy.stats import mannwhitneyu
import statsmodels.api as sm
```

In [29]:
```
#import cornell salary data
cornell_salary_df = pd.read_csv('cornell-salary-data.csv')
#convert dates to integers for year
cornell_salary_df['Starting Date'] = cornell_salary_df['Starting Date'].astype(int)
cornell_salary_df['Ending Date'] = cornell_salary_df['Ending Date'].astype(int)
#convert majors to strings
cornell_salary_df['Cornell Major'] = cornell_salary_df['Cornell Major'].astype(str)
cornell_salary_df['National Data Comparison Major'] = cornell_salary_df['National Data Compariso

#import salaries by college type
salary_college_df = pd.read_csv('salary_college_df_cleaned.csv')

#import salaries by region
salary_region_df = pd.read_csv('salary_region_df_cleaned.csv')

#import salaries by major
degree_df = pd.read_csv('degree_df_cleaned.csv')
```

Hypothesis 1

Cornell Postgraduate median salaries are greater than national postgraduate median salaries for more than 95% of majors.

We will run A/B testing to test whether Cornell Postgraduate median salaries are greater than national postgraduate median salaries. We will do a t-test to compare the means of the two samples and also a Mann–Whitney U Test to compare whether one group has larger values than the other. We will couple these A/B tests with visual representations of our data to inform the reader fully on whether Cornell postgraduate median salaries are truly greater than national postgraduate median salaries.

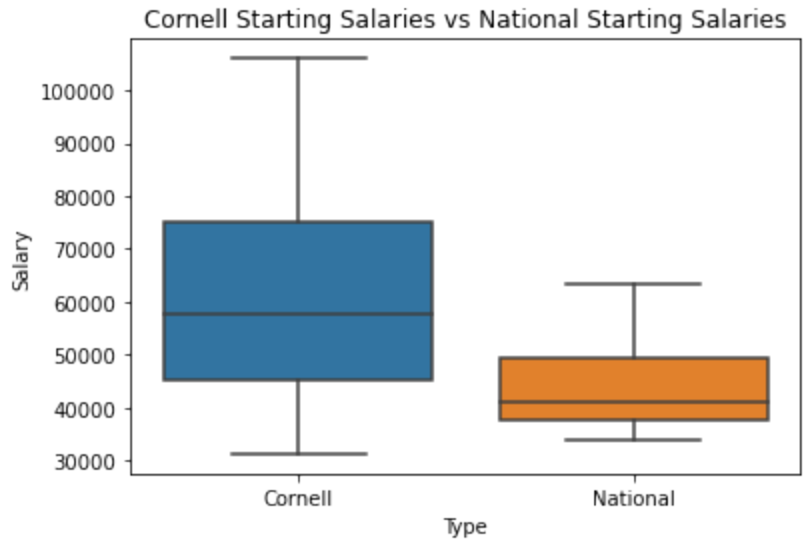
First we construct one dataset corresponding to Cornell majors and median salaries and National majors and median salaries. This dataset is called comparison_df with columns [Cornell Major, Cornell Median Salary, National Major, and National Major Median Salary]

In [30]:
```
comparison_df = pd.DataFrame()
comparison_df['Cornell Major'] = cornell_salary_df['Cornell Major']
comparison_df['Cornell Median Salary'] = cornell_salary_df['Median Salary']
comparison_df['National Major'] = cornell_salary_df['National Data Comparison Major']
comparison_df = comparison_df.sort_values('National Major')
comparison_df = comparison_df.reset_index()
comparison_df = comparison_df.drop('index',axis=1)
national_majors = comparison_df['National Major']
df_1= degree_df[degree_df['Undergraduate Major'].isin(national_majors)]
df_1 = df_1.sort_values('Undergraduate Major')
df_1 = df_1.reset_index()
df_1 = df_1.drop('index',axis=1)
comparison_df['National Major Median Salary'] = df_1['Starting Median Salary']
```

We first make a boxplot of the two dataframes.

In [31]:
```
df1 = pd.DataFrame()
df1['Salary'] = comparison_df['Cornell Median Salary']
df1['Type'] = 'Cornell'
df2 = pd.DataFrame()
df2['Salary'] = comparison_df['National Major Median Salary']
df2['Type'] = 'National'
boxplot_df = pd.concat(objs=[df1,df2])
seaborn.boxplot(x='Type',y='Salary',data=boxplot_df)
plt.title('Cornell Starting Salaries vs National Starting Salaries')
```

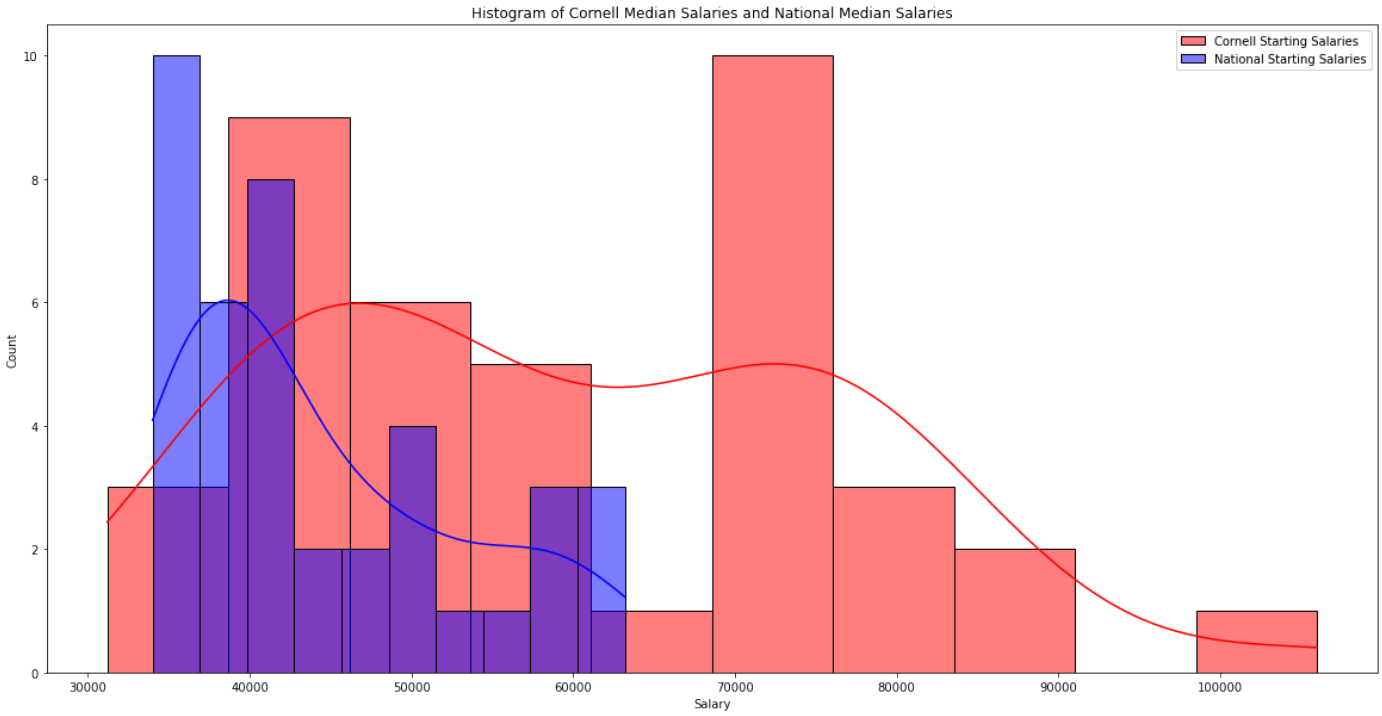
Out[31]:
```
Text(0.5, 1.0, 'Cornell Starting Salaries vs National Starting Salaries')
```



We can also make histograms and other visuals comparing Cornell Starting Salary to National Starting Salaries.

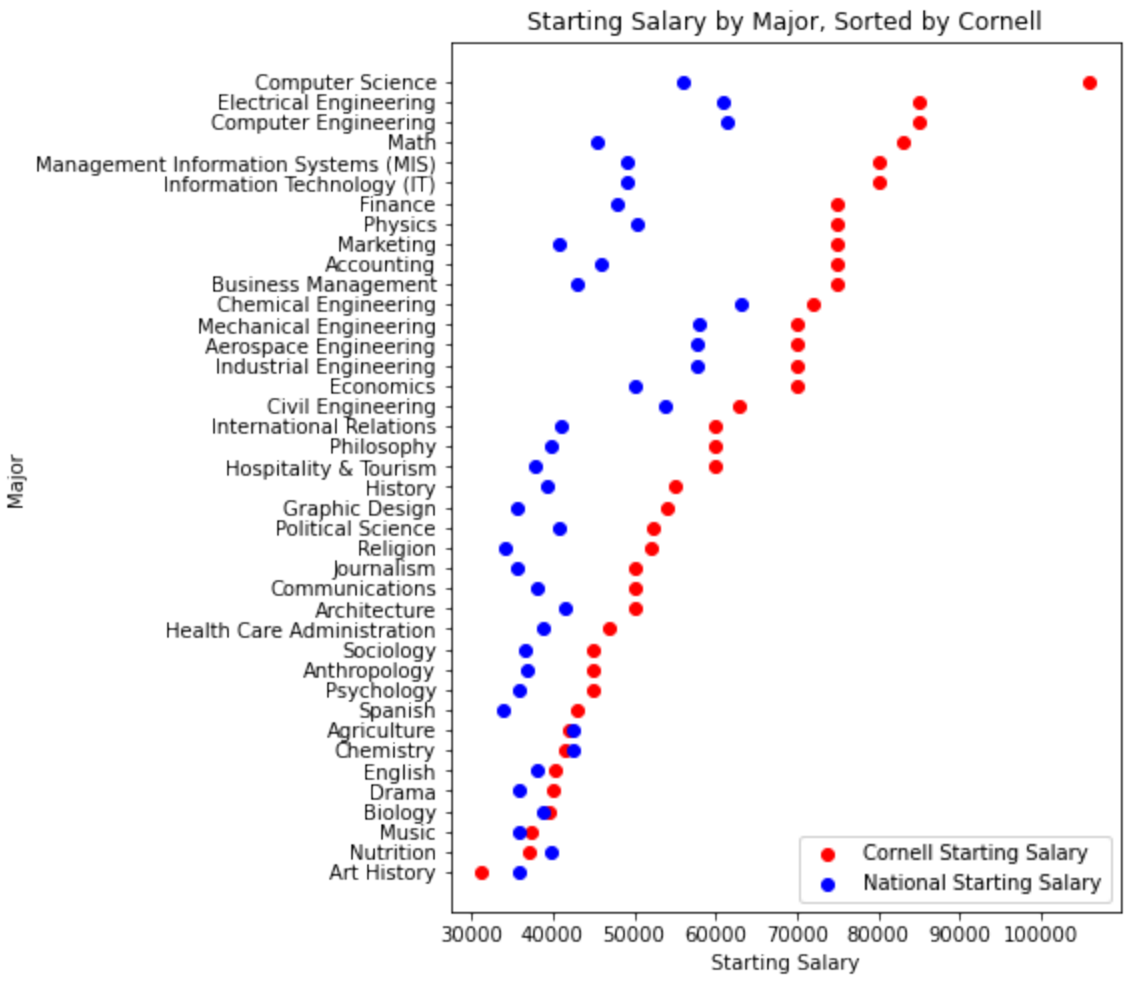
In [32]:
```
df = comparison_df
plt.figure(figsize=(20,10))
seaborn.histplot(df, x='Cornell Median Salary', kde=True,bins=10,color='Red',label='Cornell Star
seaborn.histplot(df, x='National Major Median Salary', kde=True,bins=10,color='Blue', label='Nat
plt.xlabel('Salary')
plt.ylabel('Count')
plt.legend()
plt.title('Histogram of Cornell Median Salaries and National Median Salaries')
```

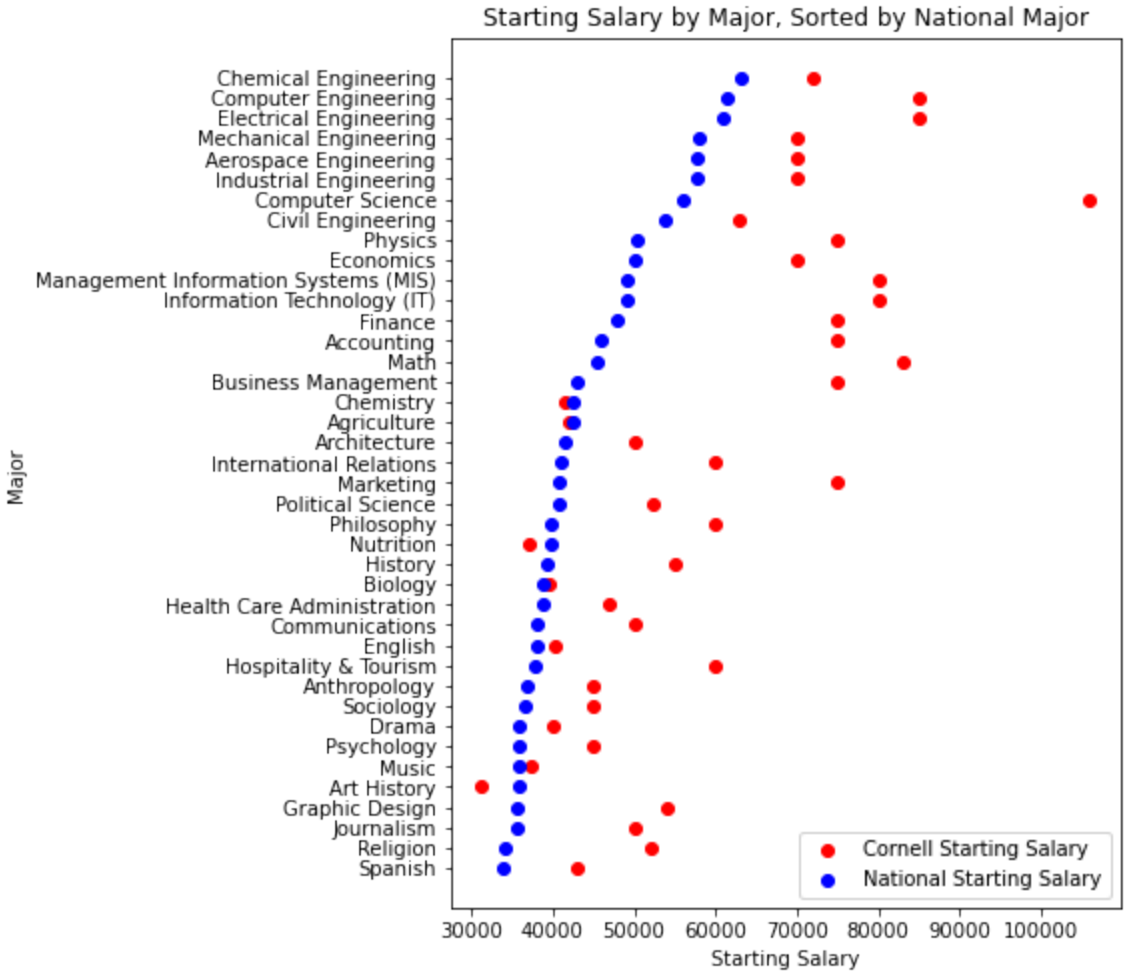
Out[32]:
```
Text(0.5, 1.0, 'Histogram of Cornell Median Salaries and National Median Salaries')
```



In [33]:
```
fig = plt.figure(figsize=(6,8))

df = comparison_df.sort_values('Cornell Median Salary').reset_index()

x = df['Cornell Median Salary']
y = df.index
labels = df['National Major']
```

```
plt.scatter(x, y, color='red', label = 'Cornell Starting Salary')
plt.yticks(y, labels)

x2 = df['National Major Median Salary']
plt.scatter(x2, y, color='blue', label = 'National Starting Salary')

plt.xlabel('Starting Salary')
plt.ylabel('Major')
plt.title('Starting Salary by Major, Sorted by Cornell')
plt.legend()
plt.show()

fig = plt.figure(figsize=(6,8))

df = comparison_df.sort_values('National Major Median Salary').reset_index()

x = df['Cornell Median Salary']
y = df.index
labels = df['National Major']

plt.scatter(x, y, color='red', label = 'Cornell Starting Salary')
plt.yticks(y, labels)

x2 = df['National Major Median Salary']
plt.scatter(x2, y, color='blue', label = 'National Starting Salary')

plt.xlabel('Starting Salary')
plt.ylabel('Major')
plt.title('Starting Salary by Major, Sorted by National Major')
plt.legend()
plt.show()
```





We can see from the graphs that overall Cornell Starting Salaries and much higher than national starting salaries. We will test the significance of this using A/B testing

T-test

We will use Welch's t-test, since the variances of the samples are different. This test compares the mean of two distributions with unequal variances.

We will set sample1 = Cornell Starting Salaries and sample2 = National Starting Salaries.

Our hypotheses are therefore:

$H_0 = \mu_1 = \mu_2$

$H_a = \mu_1 > \mu_2$

In [34]:
```
#perform t-test
# one-sided Welch's t-test with alternative = greater
stat, p_value = ttest_ind(comparison_df['Cornell Median Salary'], comparison_df['National Major
print(f"Welch's t-test: statistic={stat:.4f}, p-value={p_value:.4f}")
```

```
Welch's t-test: statistic=5.0716, p-value=0.0000
```

The p value of the test is 0, so we reject the null hypothesis. We conclude that the alternative hypothesis is true, that $\mu_1 > \mu_2$.

Mann-Whitney U test

This test will determine if one population (The cornell median starting salaries) are greater than another population (the national median starting salaries)

H0: The two populations are equal

H1: The two populations are not equal

In [35]:
```
# perform Mann Whitnet U test
stat, p_value = mannwhitneyu(comparison_df['Cornell Median Salary'], comparison_df['National Maj
print(f" Mann-Whitney U Test: statistic={stat:.4f}, p-value={p_value:.4f}")
```

```
 Mann-Whitney U Test: statistic=1252.0000, p-value=0.0000
```

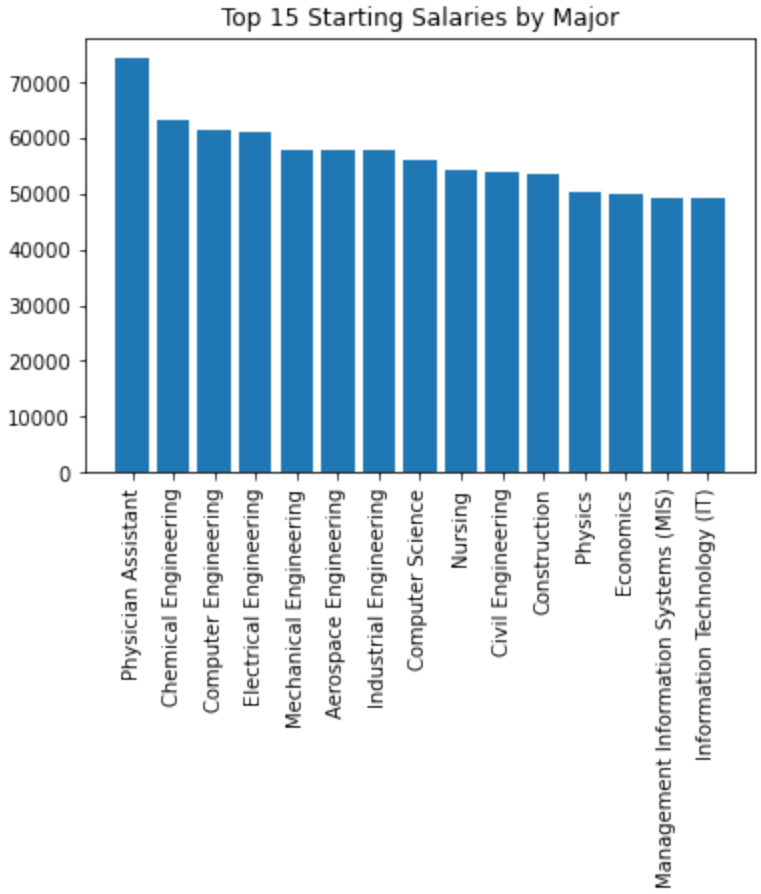
The p value of the test is 0, so we reject the null hypothesis. We conclude that the alternative hypothesis is true, that the two populations are not equal. In this case, we see that the cornell median salary population is overall greater than the national median salary population. Therefore, because of these two tests, we can strongly statistically conclude that our intuition from the graphs was correct, that median starting salaries for Cornell majors are greater than median starting salaries for National majors.

Hypothesis 2

For national data, STEM majors make more money both in starting salaries and over time than liberal arts majors

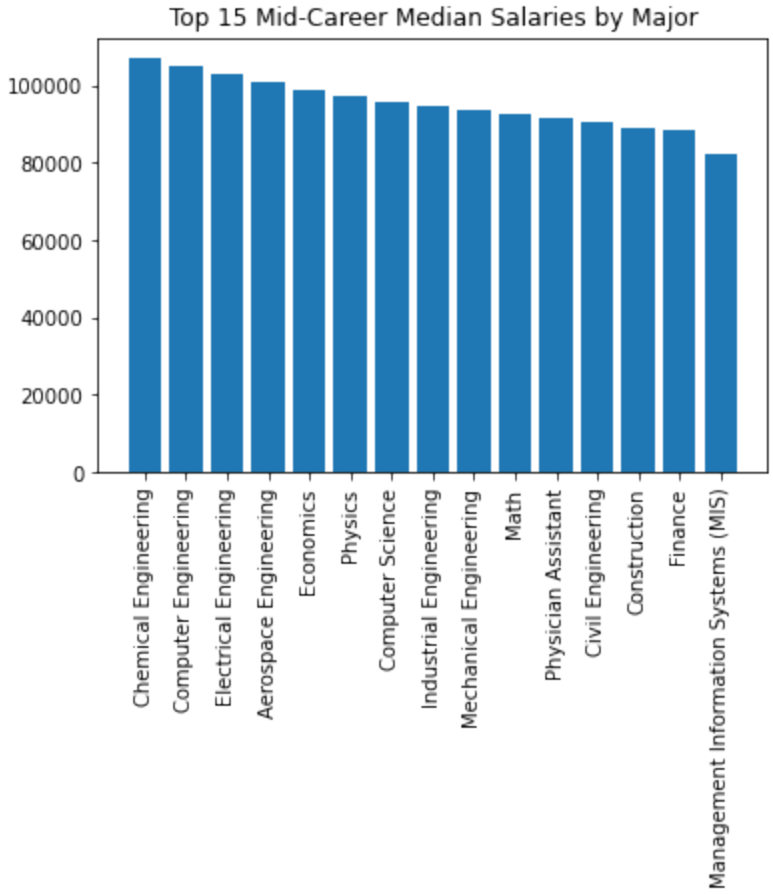
Here, we conduct an exploratory data analysis of the top 15 starting salaries by major for the national dataset. We see that the top 15 starting salaries are mostly in STEM and engineering, which informed our research questions of comparing STEM degrees to liberal arts degrees. We are including this exploratory data analysis because it informed our testing of whether STEM degrees make more money than liberal arts degrees

In [36]:
```
degree_df = degree_df.sort_values(by=['Starting Median Salary'],ascending=False)
plt.bar(degree_df['Undergraduate Major'][0:15], degree_df['Starting Median Salary'][0:15])
plt.xticks(rotation = 90)
plt.title('Top 15 Starting Salaries by Major')
plt.show()
```



Here we similarly analyze the top 15 median mid-career salaries by major, to see whether the highest earning salaries by major changed from Starting to Mid Career. We see that it did change slightly. The medical degrees are not in the top 15 anymore, although engineering degrees are. We also see that The greatest mid career median salary is higher than the greatest starting salary, indicating a general increase from starting salaries to mid career. We will use this in our analysis to see if major determines rate of increase from starting salary to mid career salary

In [37]:
```
degree_df = degree_df.sort_values(by=['Mid-Career Median Salary'],ascending=False)
plt.bar(degree_df['Undergraduate Major'][0:15], degree_df['Mid-Career Median Salary'][0:15])
plt.xticks(rotation = 90)
plt.title('Top 15 Mid-Career Median Salaries by Major')
plt.show()
```



Using Starting Median Salary and Mid-Career Median Salary to predict whether or not a major is STEM (Logistic Regression)

In [38]:
```
stem_or_nah = degree_df[['Undergraduate Major','Starting Median Salary','Mid-Career Median Salar
stem_or_nah = stem_or_nah.set_index('index')
stem_or_nah['STEM?'] = 0
stem_or_nah.iloc[[1,4,6,8,9,10,12,13,19,25,27,30,31,35,37,38,40,41,43,44],[3]]=1
stem_or_nah['Starting Median Salary'] = stem_or_nah['Starting Median Salary'] / 1000
stem_or_nah['Mid-Career Median Salary'] = stem_or_nah['Mid-Career Median Salary'] / 1000
STEM_train, STEM_test = train_test_split(stem_or_nah, test_size = 0.3, random_state = 2950)
```

We divide each salary by 1000 to gain a more useful regression equation. Inputting inputs of smaller magnitude allows us to determine the probability of being STEM given an increase of $1000 in salary, rather than a much smaller and less significant increase in salary by 1$.

We decided to use a Logistic Regression to tackle this hypothesis because we figured we could use the "Starting Median Salary" and "Mid-Career Salary" columns as our input/predictor variables and the binary column "STEM?" (which is 1 if the major is a STEM discipline, and 0 if not) as the output/predicted variable. We decided to create this model to predict whether or not a major is STEM, depending on the respective majors starting salary and mid-career salary.

In [39]:
```
def run_logistic_regression(var_names, train, test, target):
    model = LogisticRegression().fit(train[var_names], train[target])
    train_pred = model.predict(train[var_names])
    test_pred = model.predict(test[var_names])
    train_acc = metrics.accuracy_score(train[target], train_pred)
    test_acc = metrics.accuracy_score(test[target], test_pred)
    train_prec = metrics.precision_score(train[target], train_pred)
    test_prec = metrics.precision_score(test[target], test_pred)
    train_reca = metrics.recall_score(train[target], train_pred)
```

```
    test_reca = metrics.recall_score(test[target], test_pred)

    print("Predictor Variables",var_names)
    print('Coefficients', model.coef_)
    print("Intercept", model.intercept_)
    print('Train Accuracy', train_acc)
    print('Test Accuracy',test_acc)
    print('Train Precision',train_prec)
    print('Test Precision', test_prec)
    print('Train Recall', train_reca)
    print('Test Recall', test_reca)
```

In [40]:
```
run_logistic_regression(['Starting Median Salary','Mid-Career Median Salary'], STEM_train, STEM_
```

```
Predictor Variables ['Starting Median Salary', 'Mid-Career Median Salary']
Coefficients [[ 0.28838528 -0.03180128]]
Intercept [-10.94992649]
Train Accuracy 0.8
Test Accuracy 0.8
Train Precision 0.7272727272727273
Test Precision 1.0
Train Recall 0.6666666666666666
Test Recall 0.625
```

After running our Multivariable Logistic Regression by predicting on 'Starting Median Salary' and 'Mid-Career Median Salary' we were able to reach the following equation, $y \sim \text{sigmoid}(\alpha + \beta_1(x_1) + \beta_2(x_2))$ where y represents the probability that the major in question is STEM, x_1 is the starting median salary, x_2 is the mid-career median salary, α is the intercept of -11.03423542, β_1 is equal to 0.28838528, and β_2 is equal to -0.03180128.

For each $1,000 increase in starting median salary, the odds of the major being STEM are multiplied by about $1.334 (e^{0.28838528})$.

On the other hand, for each $1,000 increase in mid-career median salary, the odds of the major being STEM are multiplied by $0.969 (e^{-0.03180128})$.

Having the odds of a major being STEM increasing by a third for each time the starting salary increases by $1,000 suggests that STEM majors tend to have higher starting salaries than non-STEM majors right tw$ 1,000 suggests that mid-career salaries are harder to distinguish between STEM majors and non-STEM majors. The main conclusion from the coefficient values is that STEM majors tend to have higher starting salaries than non-STEM majors right out of school. However, this disparity becomes less significant as individuals reach mid-career as the STEM and non-STEM majors are making similar salaries on average.

Since the Train Accuracy and Test Accuracy are equally the high value of 80%, we can conclude that our regression model was mostly accurate. Also, we can conclude that we did not overfit the model (since the train and test accuracy values are the same). The Test Precision being 1.0 conveys that in our test data, all of the positive predictions were correct (true positives). The Train Precision was less accurate and had more false positives. The Train Recall being higher than the Test Recall suggests that the test set had more false negatives that the train set. Using a train test split here allows us to analyze our model's performance and potentially use this model to determine the probability of being Stem or not from other salary numbers.

We now run a statsmodels Logistic regression to easily evaluate the statistical significance of each predictors in our model.

In [41]:
```
sm_model = sm.Logit(STEM_train['STEM?'],sm.add_constant(STEM_train[['Starting Median Salary','Mi
print(sm_model.pvalues)
sm_model.summary()
```

```
Optimization terminated successfully.
         Current function value: 0.404125
         Iterations 6
const                        0.001483
Starting Median Salary       0.038273
Mid-Career Median Salary     0.601465
dtype: float64
```

Out[41]:

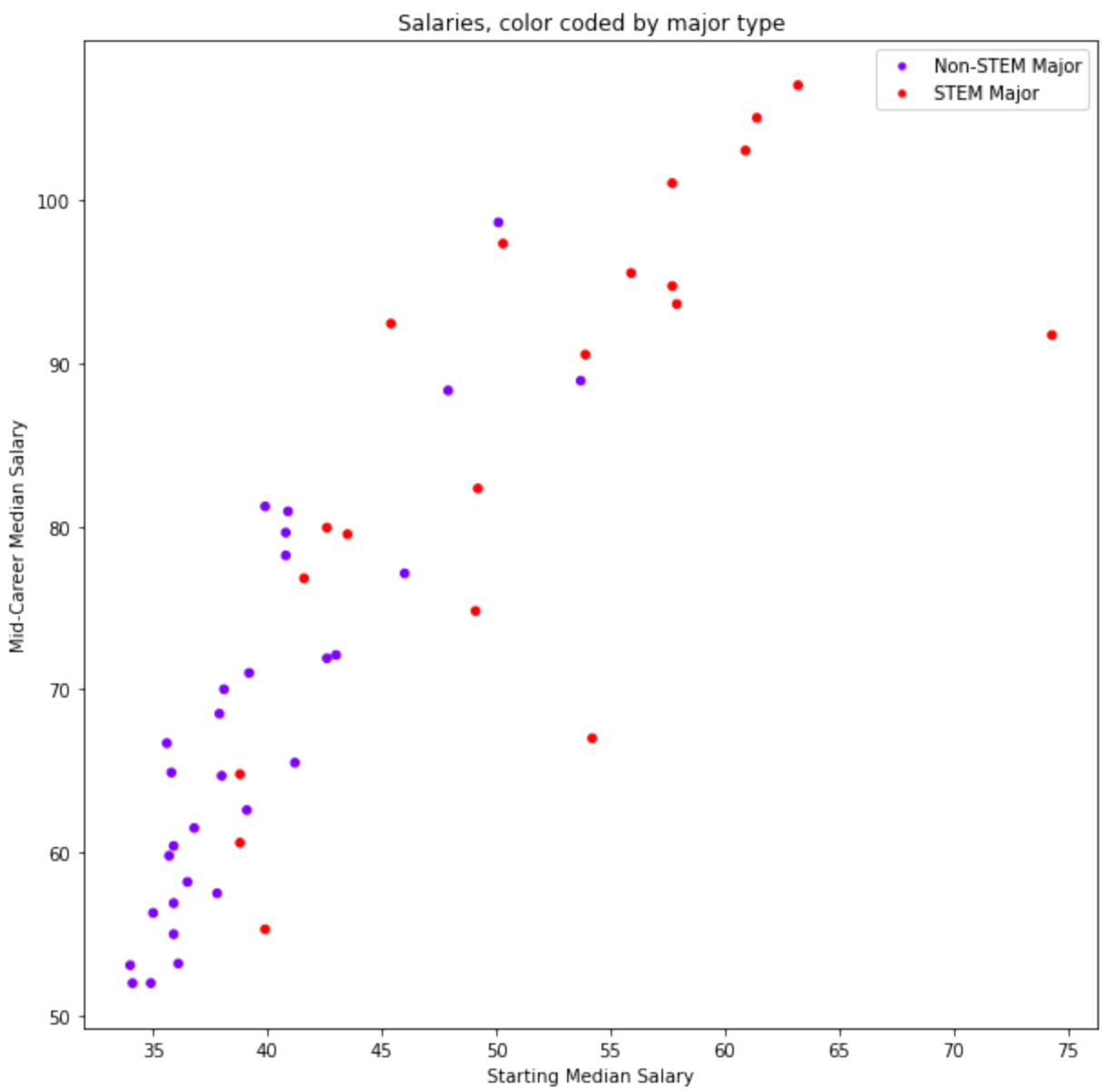
Logit Regression Results

Dep. Variable:	STEM?	**No. Observations:**	35
Model:	Logit	**Df Residuals:**	32
Method:	MLE	**Df Model:**	2
Date:	Tue, 06 Dec 2022	**Pseudo R-squ.:**	0.3714
Time:	15:02:56	**Log-Likelihood:**	-14.144
converged:	True	**LL-Null:**	-22.502
Covariance Type:	nonrobust	**LLR p-value:**	0.0002346

	coef	std err	z	P>|z|	[0.025	0.975]
const	-11.0342	3.472	-3.178	0.001	-17.840	-4.229
Starting Median Salary	0.2944	0.142	2.072	0.038	0.016	0.573
Mid-Career Median Salary	-0.0341	0.065	-0.522	0.601	-0.162	0.094

We can see from the p value of Starting Median Salary being <.05, that Starting Median Salary is a significant predictor of whether you studied a STEM major or not. However, Mid Career Median Salary is not significant with a p-value of 0.601, indicating that Mid Career Median Salary is not a significant predictor of whether you studied a STEM major or not. This echoes the results found in our Multi-Variable Logistic Regression (sklearn) model that we ran above and the coefficient values we found there.

In [42]:
```
X = np.array(stem_or_nah[['Starting Median Salary','Mid-Career Median Salary']])
y = np.array(stem_or_nah['STEM?'])

plt.figure(figsize=(10,10))
plot = plt.scatter(X[:, 0], X[:, 1], c=y, s=20,cmap='rainbow')
plt.xlabel('Starting Median Salary')
plt.ylabel('Mid-Career Median Salary')
plt.title('Salaries, color coded by major type')

maps = {"STEM Major":1,"Non-STEM Major":0}
lp = lambda i: plt.plot([],color=plot.cmap(plot.norm(i)), ms=np.sqrt(20), mec="none", label=str(
handles = [lp(i) for i in np.unique(y)]
plt.legend(handles=handles)
plt.show()
```



The findings from the Multivariable Logistic Regression above can be observed in the graph above. Ask you can see, most of the red points (STEM majors) are to the right of the majority of purple points (non-STEM major). This depicts how the starting median salaries of STEM graduates are observably higher on average than those of non-STEM graduates. However, when looking at the y-axis, you can observe that distinctions between which color points are higher than the others are harder to observe: suggesting that mid-career salary is a much worse indicator of whether or not a major was STEM or non-STEM.

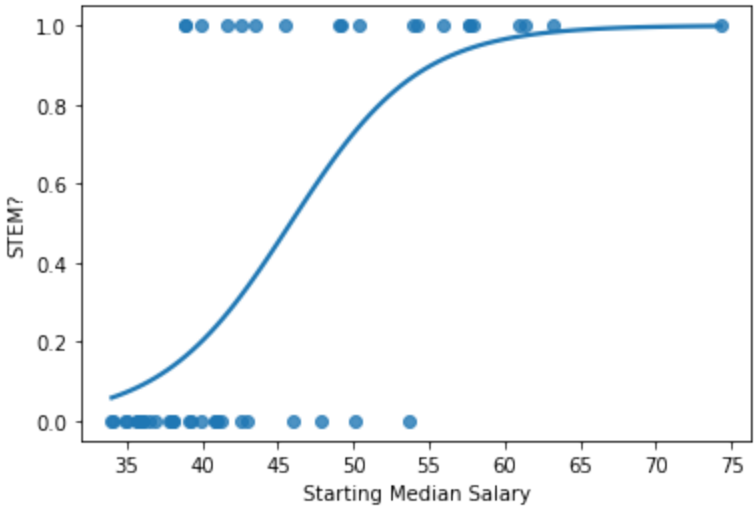
In [43]:
```
seaborn.regplot(x='Mid-Career Median Salary', y='STEM?', data=stem_or_nah, logistic=True, ci=Non
```

Out[43]:
```
<AxesSubplot:xlabel='Mid-Career Median Salary', ylabel='STEM?'>
```

This graph is interesting as it demonstrates how mid-career median salary impacts the probability of a major being STEM. For a mid-career salary of $80,000, there is approximately a 50% chance that it is a STEM major. Since the graph is pretty linear, it that as the mid-career median salary increases, the probability of it being a STEM major increases proportionally.

In [44]:
```
seaborn.regplot(x='Starting Median Salary', y='STEM?', data=stem_or_nah, logistic=True,ci=None)
```

Out[44]:
```
<AxesSubplot:xlabel='Starting Median Salary', ylabel='STEM?'>
```



This graph is much steeper than the above graph which suggests that starting median salary has a more drastic impact on whether a major is predicted to be STEM or non-STEM (as demonstrated in our LogReg model above). A major with a starting median salary of $50,000 has a 70% chance of being a STEM major. This is much more extreme than the mostly linear plot above.

To conclude, our hypothesis was partially correct. Through the use of our national dataset, we were able to observe that on average STEM majors had higher starting median salaries than non-STEM majors. This suggests that if you want to make the most money right after school you should study a STEM field.

However, as individuals get farther and farther into their careers, this salary difference is harder and harder to observe. Although both STEM and non-STEM majors make considerably more money mid-career, it is important to note that the first Logistic Regression curve suggests that as the median mid-career salary increases the odds of it being a STEM major increases as well. However, if you only care about making money farther down the line (mid-career), the importance of studying a STEM discipline is less paramount and you should be encouraged to study whatever interests you the most.

Hypothesis 3

Ivy league graduates have a greater increase in median salaries over time compared to other types of schools

In order to evaluate if Ivy League graduates have a greater increase in median salaries over time compared to other types of schools, we chose to analyze how the type of school you attend affects your starting salary and median salary. To analyze this, we chose Naive Bayes to see if starting salary and median salary were accurate predictors for the probability that you attended the type of school you attended. By using a Naive Bayes model, we can analyze the accuracy of the model to see how accurate starting salary and median salary were for predicting the type of school you went to, including types of party, liberal arts, engineering, ivy league, and state schools.

Analysis of School Type and Starting Salaries

Using Gaussian Naive Bayes to develop a prediction model

Here, we predict the probability of a student attending each school type based on starting median salary and mid-career median salary using Naive Bayes Classification. We use Naive Bayes classification because the accuracy of the model's prediction can be interpreted as the significance that school type has on your starting salary and mid career median salary. If the model is highly accurate, we can conclude that school type influences starting and mid career median salary. If the model returns a lot accuracu, we can conclude that school type does not influence starting and mid career median salary. In our analysis, we use this to specifically analyze the difference between Ivy League schools and other schools to see if median salaries over time (starting and mid career median salaries) are higher for Ivy League schools compared to other types of schools.

We start with trying to run a Naive Bayes model on all of the school types to see if our model can distinguish between school types

In [45]:
```
school_map = {"Party": 0, "State": 3, "Liberal Arts": 1, "Engineering": 4, "Ivy League": 2}
salary_college_df['School Type'] =  salary_college_df['School Type'].map(school_map)
```

We Mapped:

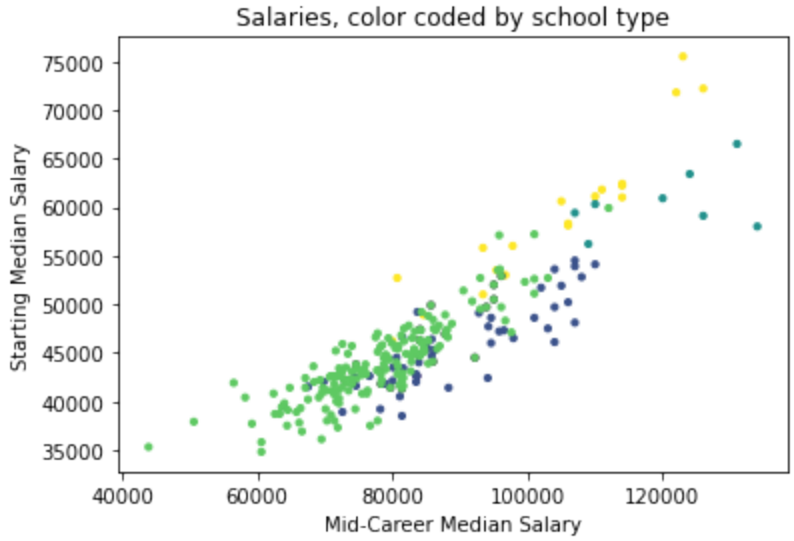
School Type	
Engineering	4
Liberal Arts	1
Ivy League	2
State	3
Party	0

We find duplicates in the data frame, because most "party" schools are also "state" schools.

In [46]:
```
X = np.array(salary_college_df[['Starting Median Salary','Mid-Career Median Salary']])
y = np.array(salary_college_df['School Type'])

plt.scatter(X[:, 1], X[:, 0], c=y, s=10)
plt.ylabel('Starting Median Salary')
```

```
plt.xlabel('Mid-Career Median Salary')
plt.title('Salaries, color coded by school type')
```

Out[46]:
```
Text(0.5, 1.0, 'Salaries, color coded by school type')
```



We see that we can only see four college types in the scatter plot. We hypothesize that many state schools are also party schools. We think that schools classified as "party" will probably have duplicates in the dataset

In [47]:
```
grouped = salary_college_df.groupby('School Name').count().reset_index()
duplicate_schools = list(grouped[grouped['School Type']>1]['School Name'])
```

Lets look more into why these schools are duplicate. This is another form of cleaning that we discovered was necessary during analysis.

In [48]:
```
duplicates_df = salary_college_df[salary_college_df['School Name'].isin(duplicate_schools)].sort
duplicates_df = duplicates_df[duplicates_df['School Type']!=0]
```

We see that all duplicates are school type 3 and 4, which corresponds to state and party. Except 'Randolph Macon College' which is liberal arts and party. We choose to drop rows from the original dataframe that have duplicate categories, preserving the "Party" category and ignoring the duplicate category. We want to analyze the impact of "party" schools specifically later in our project, which is why we must preserve the "Party" representation and drop the duplicte representation as "State" or "Liberal Arts"

In [49]:
```
drop_indicies = list(duplicates_df.index)
#make a copy just in case our drop fails
salary_college_df1 = salary_college_df
salary_college_df1 = salary_college_df1.drop(index=drop_indicies)

#check that there are no duplicates
grouped = salary_college_df1.groupby('School Name').count().reset_index()
grouped[grouped['School Type']>1]
```

Out[49]:

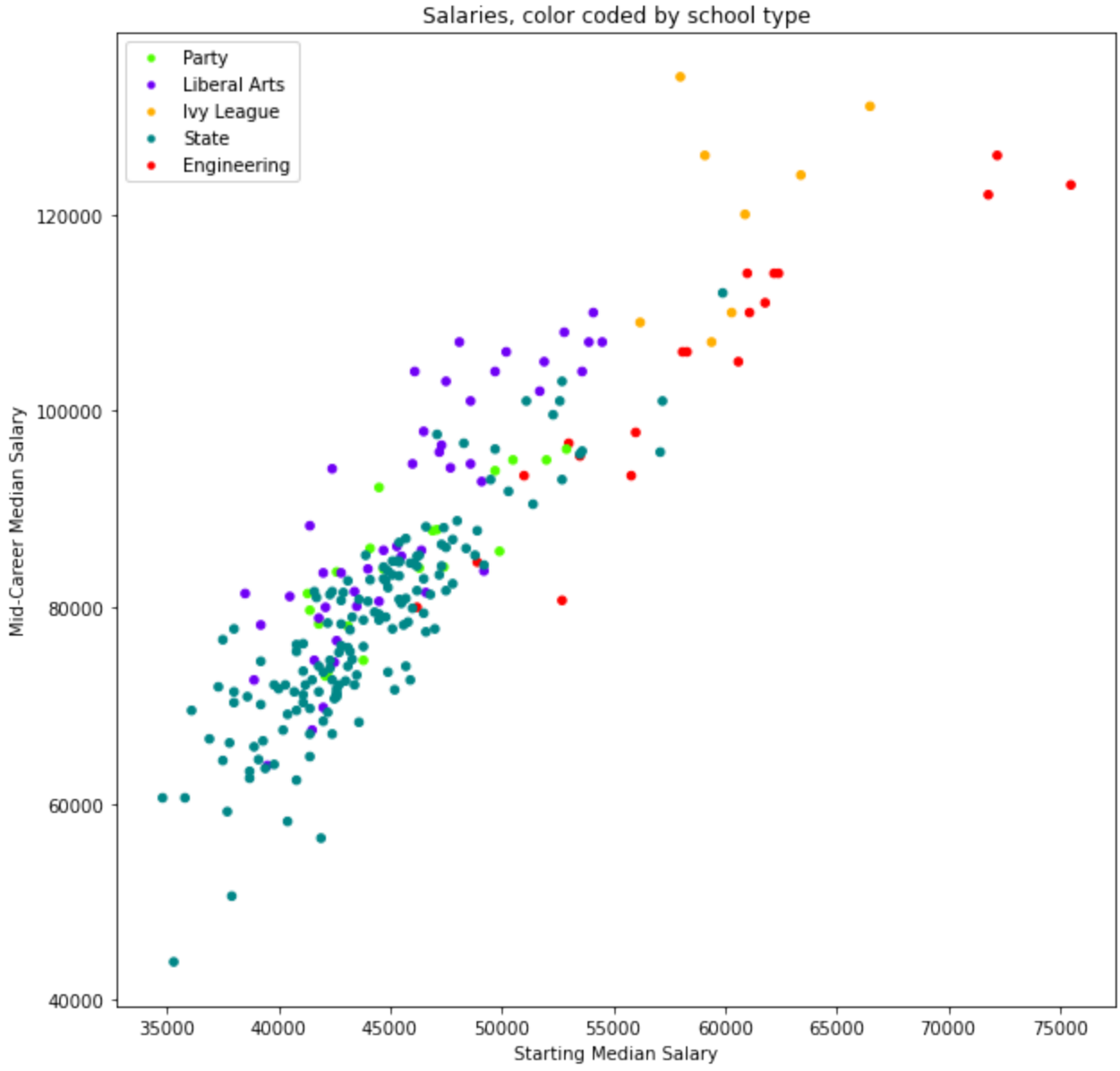
School Name	School Type	Starting Median Salary	Mid-Career Median Salary	Mid-Career 10th Percentile Salary	Mid-Career 25th Percentile Salary	Mid-Career 75th Percentile Salary	Mid-Career 90th Percentile Salary	Percent change from Starting to Mid-Career Salary

We see that there are no more duplicates (we removed them all), so we can reassign our original dataframe to be the new dataframe and replot the scatter plot using our original map. We hope to see 5 distinct colors

in the scatter plot

In [50]:
```
school_map = {"Party": 0, "State": 3, "Liberal Arts": 1, "Engineering": 4, "Ivy League": 2}
```

In [51]:
```
salary_college_df= salary_college_df1

X = np.array(salary_college_df[['Starting Median Salary','Mid-Career Median Salary']])
y = np.array(salary_college_df['School Type'])

plt.figure(figsize=(10,10))
plot = plt.scatter(X[:, 0], X[:, 1], c=y, s=20,cmap='prism_r')
plt.xlabel('Starting Median Salary')
plt.ylabel('Mid-Career Median Salary')
plt.title('Salaries, color coded by school type')

#code for legend found online
lp = lambda i: plt.plot([],color=plot.cmap(plot.norm(i)), ms=np.sqrt(20), mec="none", label=str(
handles = [lp(i) for i in np.unique(y)]
plt.legend(handles=handles)
plt.show()
```



We can see that the Ivy League (Orange) and Engineering schools (red) tend to have the higher Starting and Mid-Career Salaries in general, compared to party, state, and liberal arts schools.

Run Naive Bayes Classification

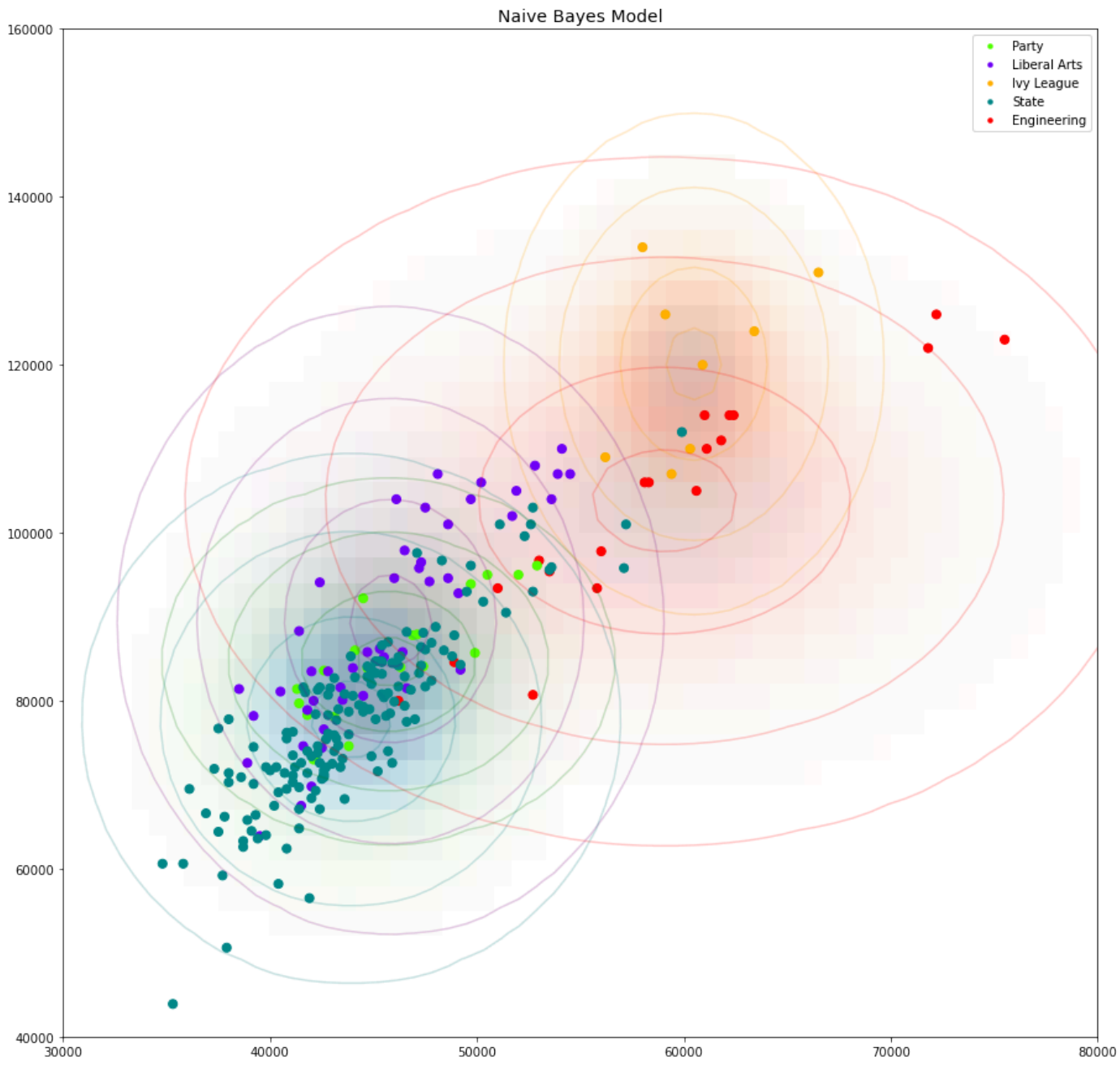
In [52]:
```
from sklearn.naive_bayes import GaussianNB


X = np.array(salary_college_df[['Starting Median Salary','Mid-Career Median Salary']])
y = np.array(salary_college_df['School Type'])

X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=.1,random_state=2950)
model = GaussianNB()
model.fit(X_train, y_train)
print("Naive Bayes score: ",model.score(X_test, y_test))
```

```
Naive Bayes score:  0.56
```

Our model has a score of 0.56. This is pretty good but not perfect. We hypothesize this is because overlapping data points based on school type as seen in our scatter plot. We conclude that school type across all school types is not a great predictor of mid career and starting salaries over time. We will plot the clusters below to obtain an even better visual of the overlap to tune our model.

In [53]:
```
fig, ax = plt.subplots(figsize=(15,15))
ax.scatter(X[:, 0], X[:, 1], c=y, s=50,cmap='prism_r')
ax.set_title('Naive Bayes Model', size=14)

ylim = (40000, 160000)
xlim = (30000, 80000)

xg = np.linspace(xlim[0], xlim[1], 60)
yg = np.linspace(ylim[0], ylim[1], 40)
xx, yy = np.meshgrid(xg, yg)
Xgrid = np.vstack([xx.ravel(), yy.ravel()]).T

for label, color in enumerate(['green', 'purple', 'orange', 'teal', 'red']):
    mask = (y == label)
    mu, std = X[mask].mean(0), X[mask].std(0)
    P = np.exp(-0.5 * (Xgrid - mu) ** 2 / std ** 2).prod(1)
    Pm = np.ma.masked_array(P, P < 0.03)


    if color=='green':
        ax.pcolorfast(xg, yg, Pm.reshape(xx.shape), alpha=0.25, cmap='Greens')
    if color=='orange':
        ax.pcolorfast(xg, yg, Pm.reshape(xx.shape), alpha=0.25, cmap='Oranges')
    if color=='teal':
        ax.pcolorfast(xg, yg, Pm.reshape(xx.shape), alpha=0.25, cmap='GnBu')
    if color=='red':
        ax.pcolorfast(xg, yg, Pm.reshape(xx.shape), alpha=0.15, cmap='Reds')
    if color=='purple':
        ax.pcolorfast(xg, yg, Pm.reshape(xx.shape), alpha=0.25, cmap='Purples')
    ax.contour(xx, yy, P.reshape(xx.shape),
               levels=[0.01, 0.1, 0.5, 0.9],
               colors=color, alpha=0.2)

#code for legend found online
lp = lambda i: plt.plot([],color=plot.cmap(plot.norm(i)), ms=np.sqrt(20), mec="none", label=str(
handles = [lp(i) for i in np.unique(y)]
ax.legend(handles=handles)
ax.set(xlim=xlim, ylim=ylim)
```

Out[53]:
```
[(30000.0, 80000.0), (40000.0, 160000.0)]
```



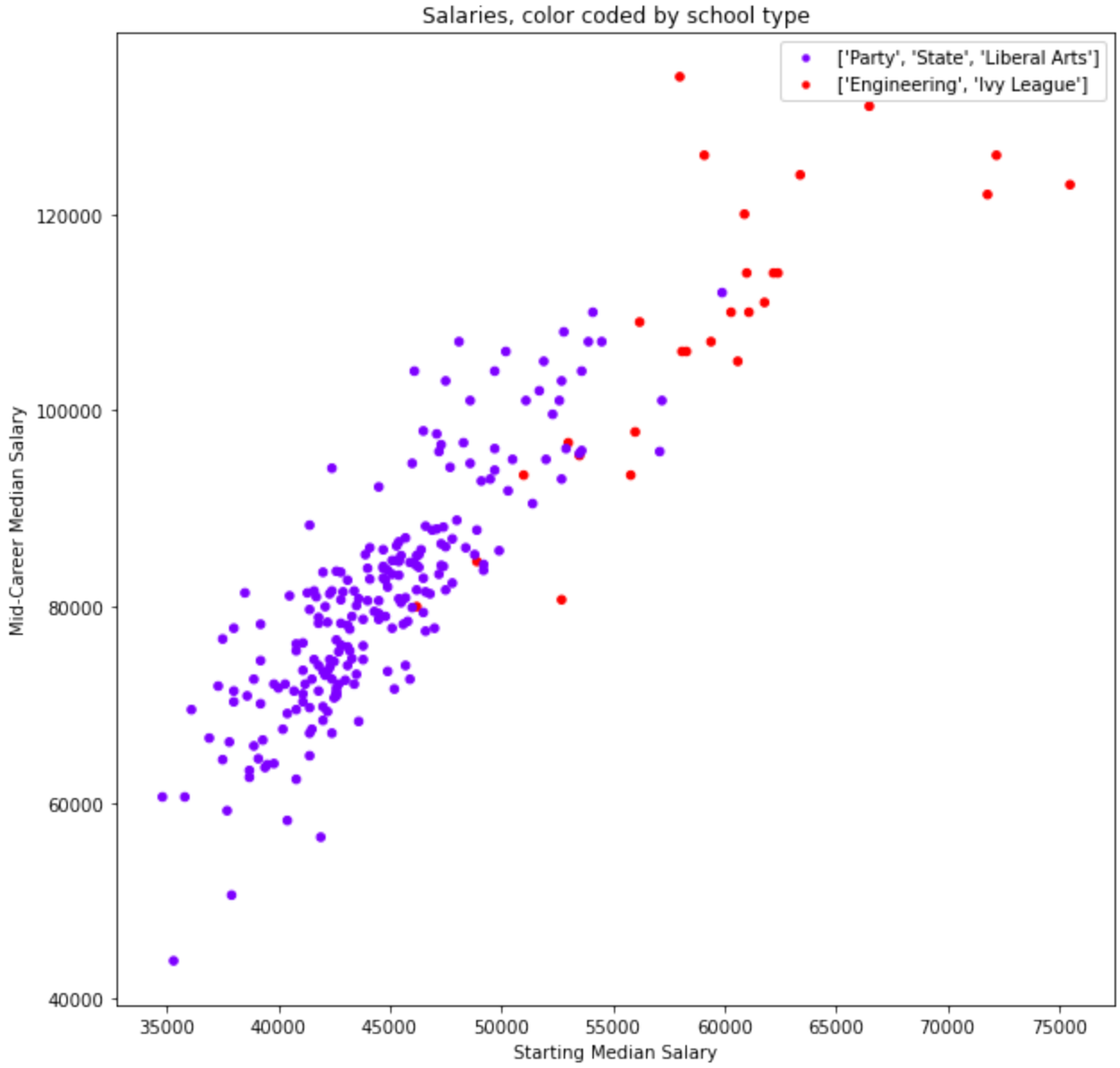
We see here that we have developed clusters for different school types. Due to the highly overlaping data for mid career median salaries and starting median salaries, we see that there is not much difference between the clusters. We will further investigate by considering Engineering and Ivy League Schools vs Liberal Arts, Party, and State Schools. This is because we saw in the clustering graph that the party, state, and liberal arts clusters overlap and the engineering and ivy league schools overlap.

We map Engineering and Ivy League Schools to 1 and Liberal Arts, Party, and State Schools to 0 to compare Engineering/Ivy League and Party/State/Liberal Arts schools. If this model returns a higher score we can conclude that Mid Career Median Salary and Starting Salary can predict a difference between these two groups of school types. Because we see in the graph that engineering and Ivy League have higher salaries, we would conclude that engineering and ivy league graduates make more money over time.

In [54]:
```
new_map = {2:1,4:1,0:0,3:0,1:0}
salary_college_df['School Type'] =  salary_college_df['School Type'].map(new_map)
```

In [55]:
```
X = np.array(salary_college_df[['Starting Median Salary','Mid-Career Median Salary']])
y = np.array(salary_college_df['School Type'])

plt.figure(figsize=(10,10))
```

```
plot = plt.scatter(X[:, 0], X[:, 1], c=y, s=20,cmap='rainbow')
plt.xlabel('Starting Median Salary')
plt.ylabel('Mid-Career Median Salary')
plt.title('Salaries, color coded by school type')
school_map = {"Party": 0, "State": 0, "Liberal Arts": 0, "Engineering": 1, "Ivy League": 1}
#code for legend found online
lp = lambda i: plt.plot([],color=plot.cmap(plot.norm(i)), ms=np.sqrt(20), mec="none", label=str(
handles = [lp(i) for i in np.unique(y)]
plt.legend(handles=handles)
plt.show()
```



In [56]:

```
X = np.array(salary_college_df[['Starting Median Salary','Mid-Career Median Salary']])
y = np.array(salary_college_df['School Type'])

X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=.1,random_state=2950)
model = GaussianNB()
model.fit(X_train, y_train)
print("Naive Bayes score: ",model.score(X_test, y_test))
```

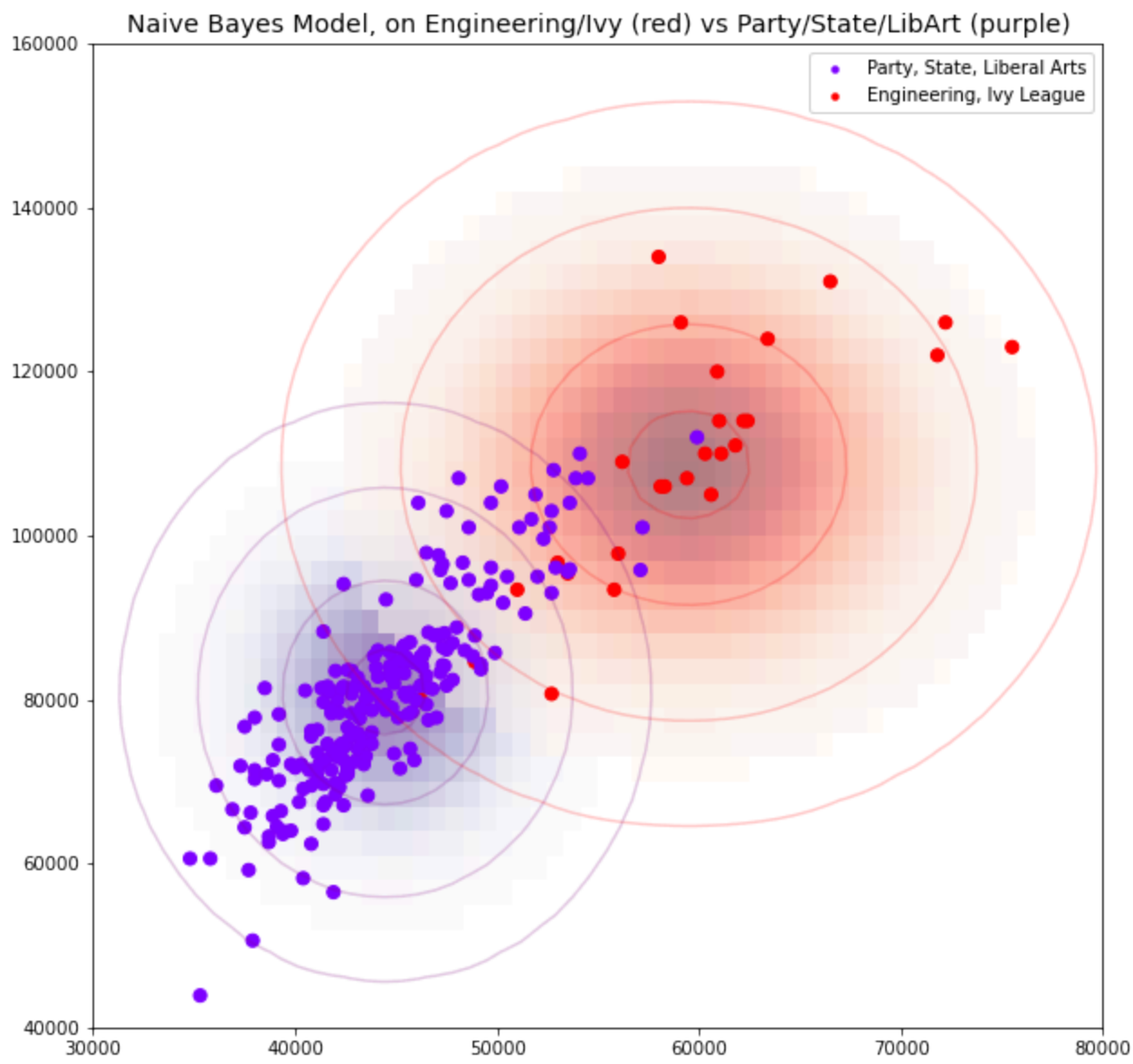
```
Naive Bayes score:  0.88
```

This new model has a much higher score of 0.88. We are more pleased and see that Engineering and Ivy League are a good predictor of high starting and mid career median salaries, while state, party, and liberal arts schools are a good predictor of lower salaries. We conclude that school type is a predictor of salaries over time, both mid career and starting salaries, for Ivy League and Engineering vs Party, State, and Liberal

Arts. To gain a higher salary in both mid career and starting, you should attend an engineering school or an Ivy League Institution.

In [58]:
```
fig, ax = plt.subplots(figsize=(10,10))
ax.scatter(X[:, 0], X[:, 1], c=y, s=50,cmap='rainbow')
ax.set_title('Naive Bayes Model, on Engineering/Ivy (red) vs Party/State/LibArt (purple)', size=

ylim = (40000, 160000)
xlim = (30000, 80000)

xg = np.linspace(xlim[0], xlim[1], 60)
yg = np.linspace(ylim[0], ylim[1], 40)
xx, yy = np.meshgrid(xg, yg)
Xgrid = np.vstack([xx.ravel(), yy.ravel()]).T

for label, color in enumerate(['purple', 'red']):
    mask = (y == label)
    mu, std = X[mask].mean(0), X[mask].std(0)
    P = np.exp(-0.5 * (Xgrid - mu) ** 2 / std ** 2).prod(1)
    Pm = np.ma.masked_array(P, P < 0.03)
    if color=='purple':
        ax.pcolorfast(xg, yg, Pm.reshape(xx.shape), alpha=0.5, cmap='Purples')
    if color=='red':
        ax.pcolorfast(xg, yg, Pm.reshape(xx.shape), alpha=0.5, cmap='Reds')

    ax.contour(xx, yy, P.reshape(xx.shape),
               levels=[0.01, 0.1, 0.5, 0.9],
               colors=color, alpha=0.2)

#code for legend found online
lp = lambda i: plt.plot([],color=plot.cmap(plot.norm(i)), ms=np.sqrt(20), mec="none", label=str(
handles = [lp(i) for i in np.unique(y)]
ax.legend(handles=handles)
ax.set(xlim=xlim, ylim=ylim)
```

Out[58]:
```
[(30000.0, 80000.0), (40000.0, 160000.0)]
```

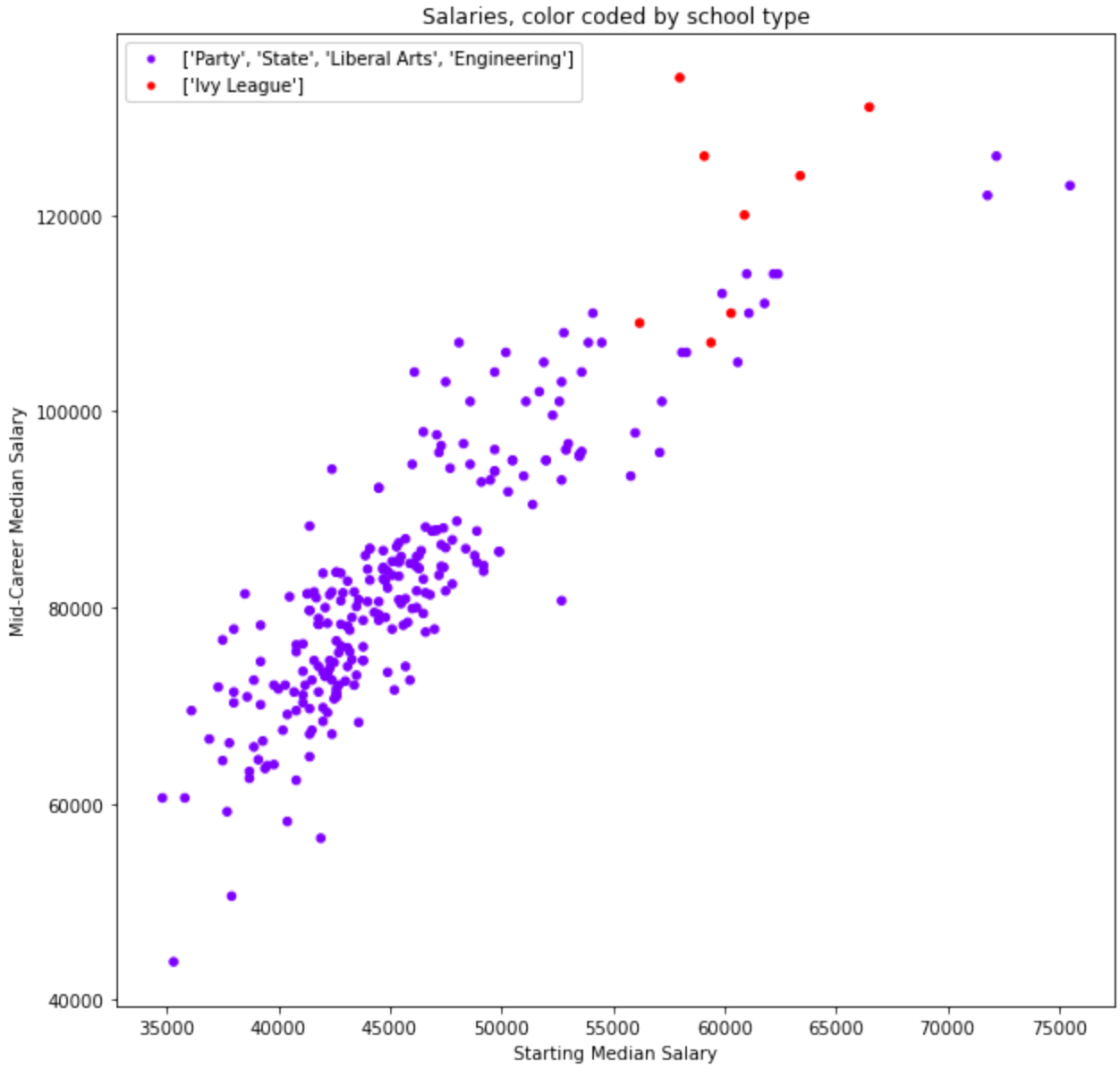


We conclude that school type is a predictor of salaries over time, both mid career and starting salaries, for Ivy League and Engineering vs Party, State, and Liberal Arts. Ivy league is not the only significant predictor of higher salaries in both mid career and starting. This also extend to engineering. To gain a higher salary in both mid career and starting, you should attend an engineering school or an Ivy League Institution.

We can now run the same classification on Ivy League vs Non Ivy League to determine whether Ivy League is even more significant than Ivy League or Engineering.

In [59]:
```
#reimport dataframe to remap between Ivy vs Non Ivy
salary_college_df = pd.read_csv('salary_college_df_cleaned.csv')
school_map = {"Party": 0, "State": 0, "Liberal Arts": 0, "Engineering": 0, "Ivy League": 1}
salary_college_df['School Type'] =  salary_college_df['School Type'].map(school_map)

X = np.array(salary_college_df[['Starting Median Salary','Mid-Career Median Salary']])
y = np.array(salary_college_df['School Type'])

plt.figure(figsize=(10,10))
plot = plt.scatter(X[:, 0], X[:, 1], c=y, s=20,cmap='rainbow')
plt.xlabel('Starting Median Salary')
plt.ylabel('Mid-Career Median Salary')
plt.title('Salaries, color coded by school type')
school_map = {"Party": 0, "State": 0, "Liberal Arts": 0, "Engineering": 0, "Ivy League": 1}
#code for legend found online
lp = lambda i: plt.plot([],color=plot.cmap(plot.norm(i)), ms=np.sqrt(20), mec="none", label=str(
handles = [lp(i) for i in np.unique(y)]
```

```
plt.legend(handles=handles)
plt.show()
```



It looks like Ivy League is a good predictor of Mid Career Median Salaries, but not Starting Salaries. We run Bayes to confirm. We will test the score of the Bayes model on one predictor (Mid Career Median Salary) and Two Predictors (Starting and Mid Career Median Salary) to compare.

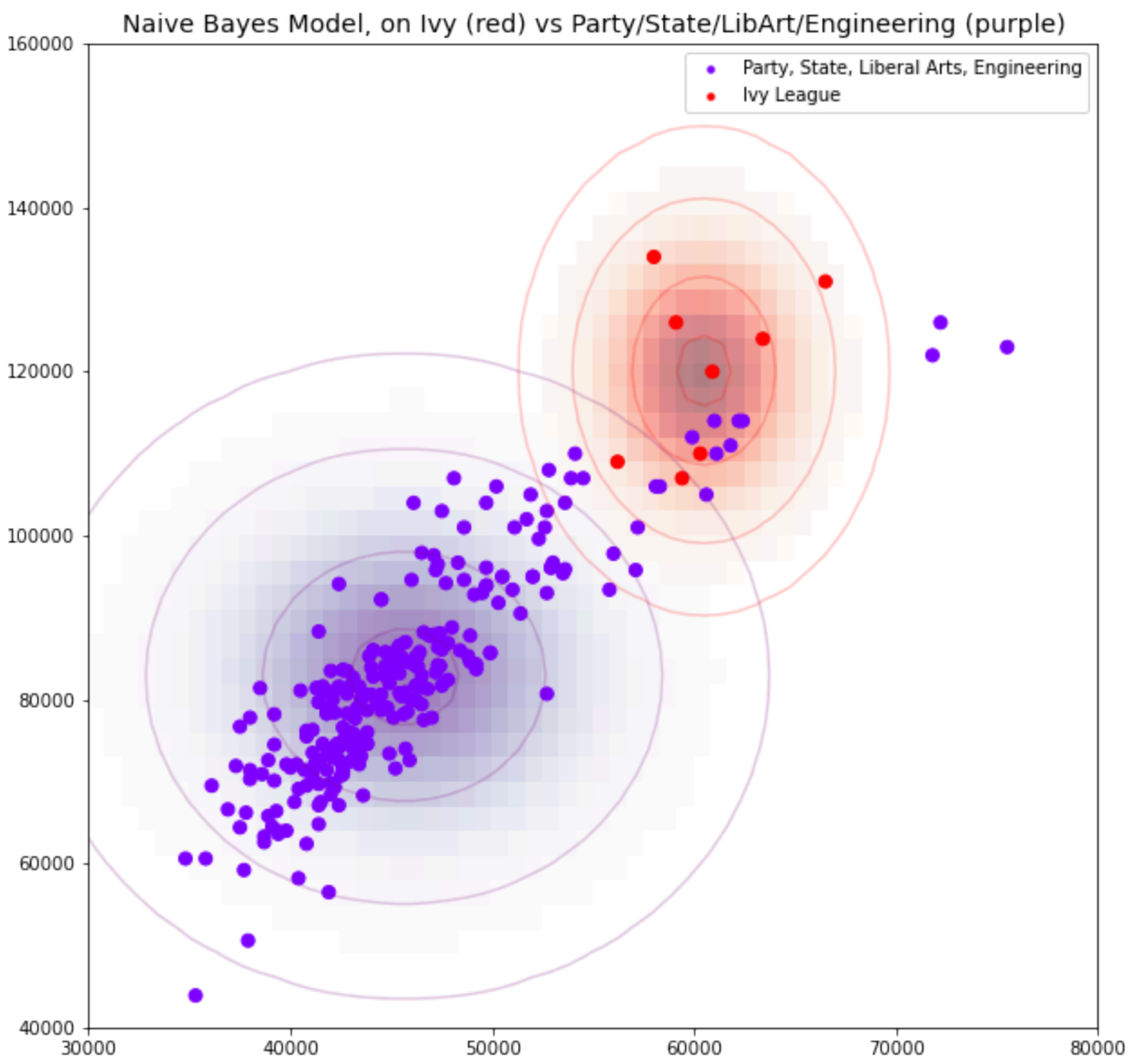
In [60]:
```
X = np.array(salary_college_df[['Starting Median Salary','Mid-Career Median Salary']])
y = np.array(salary_college_df['School Type'])

X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=.2,random_state=2950)
model = GaussianNB()
model.fit(X_train, y_train)
print("Naive Bayes score for Starting and Mid on Ivy League: ",model.score(X_test, y_test))

X1 = np.array(salary_college_df['Mid-Career Median Salary'])
y1 = np.array(salary_college_df['School Type'])

X_train, X_test, y_train, y_test = train_test_split(X1,y1,test_size=.2,random_state=2951)
model = GaussianNB()
model.fit(X_train.reshape(-1, 1), y_train)
print("Naive Bayes score for just Mid Career Salary on Ivy League: ",model.score(X_test.reshape(
```

```
Naive Bayes score for Starting and Mid on Ivy League:  0.9629629629629629
Naive Bayes score for just Mid Career Salary on Ivy League:  0.9814814814814815
```

We see that the scores for both models are very high, however the score for the model only containing Mid Career median salary is higher. Therefore, we conclude that starting salary and mid career median salary are both heavily influenced by Ivy League attendance, but mid career median salary is influenced more by Ivy League attendance. The score for these models are extremely high. We now know that a high starting salary and a especially high mid career median salary are very strong predictors of attending an ivy league institution.

In [61]:
```
fig, ax = plt.subplots(figsize=(10,10))
ax.scatter(X[:, 0], X[:, 1], c=y, s=50,cmap='rainbow')
ax.set_title('Naive Bayes Model, on Ivy (red) vs Party/State/LibArt/Engineering (purple)', size=

ylim = (40000, 160000)
xlim = (30000, 80000)

xg = np.linspace(xlim[0], xlim[1], 60)
yg = np.linspace(ylim[0], ylim[1], 40)
xx, yy = np.meshgrid(xg, yg)
Xgrid = np.vstack([xx.ravel(), yy.ravel()]).T

for label, color in enumerate(['purple', 'red']):
    mask = (y == label)
    mu, std = X[mask].mean(0), X[mask].std(0)
    P = np.exp(-0.5 * (Xgrid - mu) ** 2 / std ** 2).prod(1)
    Pm = np.ma.masked_array(P, P < 0.03)
    if color=='purple':
        ax.pcolorfast(xg, yg, Pm.reshape(xx.shape), alpha=0.5, cmap='Purples')
    if color=='red':
        ax.pcolorfast(xg, yg, Pm.reshape(xx.shape), alpha=0.5, cmap='Reds')

    ax.contour(xx, yy, P.reshape(xx.shape),
               levels=[0.01, 0.1, 0.5, 0.9],
               colors=color, alpha=0.2)

#code for legend found online
lp = lambda i: plt.plot([],color=plot.cmap(plot.norm(i)), ms=np.sqrt(20), mec="none", label=str(
handles = [lp(i) for i in np.unique(y)]
ax.legend(handles=handles)
ax.set(xlim=xlim, ylim=ylim)
```

Out[61]:
```
[(30000.0, 80000.0), (40000.0, 160000.0)]
```



Based on our Naive Bayes Classification, we conclude that attending an Ivy League school significantly influences earning a high starting salary and a high mid career median salary.

For further analysis, we can run a logistic regression to gain the significance level of Mid Career and Starting Salaries predicting whether school type attended is Ivy League.

In [75]:
```
salary_college_df = pd.read_csv('salary_college_df_cleaned.csv')
school_map = {"Party": 0, "State": 0, "Liberal Arts": 0, "Engineering": 0, "Ivy League": 1}
salary_college_df['School Type'] =  salary_college_df['School Type'].map(school_map)

sm_model = sm.Logit(salary_college_df['School Type'], sm.add_constant(salary_college_df[['Starti
print(sm_model.pvalues)
sm_model.summary()
```

```
Optimization terminated successfully.
         Current function value: 0.048996
         Iterations 11
const                       0.000307
Starting Median Salary      0.131575
Mid-Career Median Salary    0.004007
dtype: float64
```

Out[75]:

Logit Regression Results

Dep. Variable:	School Type	**No. Observations:**	269
Model:	Logit	**Df Residuals:**	266
Method:	MLE	**Df Model:**	2
Date:	Tue, 06 Dec 2022	**Pseudo R-squ.:**	0.6339
Time:	13:45:15	**Log-Likelihood:**	-13.180
converged:	True	**LL-Null:**	-36.002
Covariance Type:	nonrobust	**LLR p-value:**	1.226e-10

	coef	std err	z	P>|z|	[0.025	0.975]
const	-24.9675	6.918	-3.609	0.000	-38.526	-11.409
Starting Median Salary	-0.0002	0.000	-1.508	0.132	-0.000	5.32e-05
Mid-Career Median Salary	0.0003	0.000	2.878	0.004	9.6e-05	0.001

Possibly complete quasi-separation: A fraction 0.35 of observations can be perfectly predicted. This might indicate that there is complete quasi-separation. In this case some parameters will not be identified.

We can see that this Logistic regression has a Pseudo R-squared of 0.6339. This is a relatively good value. Additionally, by looking at the p values, we see that the Starting median salary has a p value is 0.132, which is not significant to the 95% significance level. However, Mid Career Median Salary has a p value of 0.004, which means that it is significant at the 95% signfiicance level. From this, we conclude that attending an Ivy League School highly influences your Mid Career median salary, but does not influence starting salary as strongly. Our conclusion of this hypothesis is that attending an Ivy League school will increase your median salary over time, soecifically influencing your mid-career median salary very strongly. This has been proven through Naive Bayes and Logistic regression.

Hypothesis 4

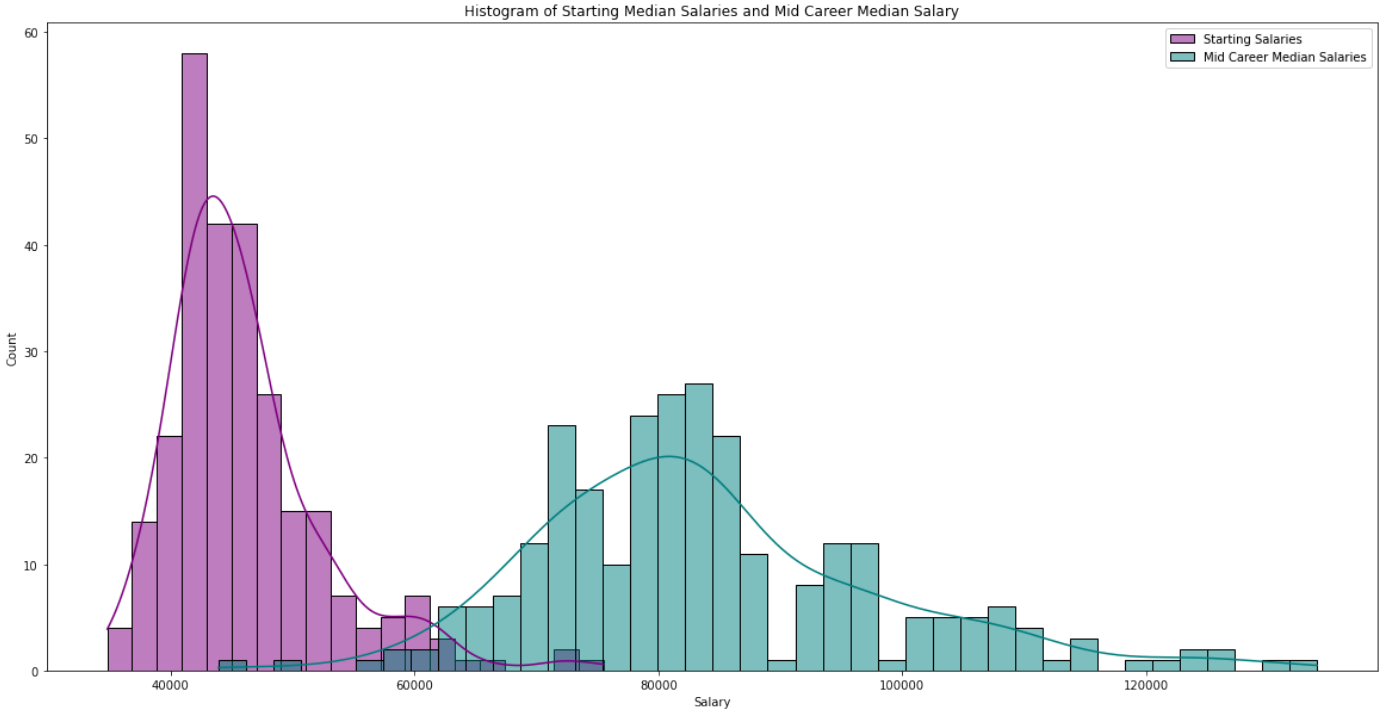
Salaries increase over time and can be predicted based on starting salary.

Starting Salary vs Mid Career Salary Visualizations

First, we perform exploratory analysis on the distribution of starting and mid career salaries.

In [62]:
```
df = salary_college_df
plt.figure(figsize=(20,10))
seaborn.histplot(df, x='Starting Median Salary', kde=True,color='Purple',bins=20,label='Starting
seaborn.histplot(df, x='Mid-Career Median Salary', kde=True,color='Teal',bins=40, label='Mid Car
plt.xlabel('Salary')
plt.ylabel('Count')
plt.legend()
plt.title('Histogram of Starting Median Salaries and Mid Career Median Salary')
```

Out[62]:
```
Text(0.5, 1.0, 'Histogram of Starting Median Salaries and Mid Career Median Salary')
```



Here, the histogram analysis of starting salaries to mid career median salaries shows that on average, it takes time to accumulate a higher salary. Starting Salaries do not indicate median salaries mid career, and often many people have to wait many years to make a lot of money. Based on this histogram, we predict that salaries increase over time.

Run Linear Regression

In order to determine if mid career salary and starting salary are correlated and if mid career salary can be predicted by starting salary, we run a linear regression model. We are using linear regression in order to observe the relationship between the Starting Median Salary and Mid-Career Median Salary. If we find that Starting Median Salary is a good predictor for Mid-Career Median Salary, we will be able to conclude that a graduate's starting median salary will have an effect on their mid-career salary.

In [63]:
```
y = salary_college_df['Mid-Career Median Salary']
X = salary_college_df[['Starting Median Salary']]

model = LinearRegression()
model.fit(X,y)
print('Coefficient:', model.coef_)
print('Intercept', model.intercept_)
print('Model Score:', model.score(X,y))
```

```
Coefficient: [1.98902892]
Intercept -7699.040771150845
Model Score: 0.7915623563069352
```

The coefficient is about 2. Therefore, we can see that when the Starting Median Salary goes up by a factor of 1, the Mid-Career Median Salary is predicted to go up by about a factor of 2. The intercept tells us that when the starting salary is 0, which does not happen in the data, then the mid career salary is predicted to be -$8279.46. This is not possible, because a salary cannot be negative. The model score of approximately 0.8 tells us that the model is very accurate. The best possible score is 1.0.

In [64]:
```
model=sm.OLS(y,X)
reg=model.fit()
reg.summary()
```

Out[64]:

OLS Regression Results

Dep. Variable:	Mid-Career Median Salary	**R-squared (uncentered):**	0.994
Model:	OLS	**Adj. R-squared (uncentered):**	0.994
Method:	Least Squares	**F-statistic:**	4.409e+04
Date:	Tue, 06 Dec 2022	**Prob (F-statistic):**	2.34e-299
Time:	15:06:01	**Log-Likelihood:**	-2748.2
No. Observations:	269	**AIC:**	5498.
Df Residuals:	268	**BIC:**	5502.
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Starting Median Salary	1.8251	0.009	209.967	0.000	1.808	1.842

Omnibus:	20.924	**Durbin-Watson:**	0.796
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	39.544
Skew:	0.424	**Prob(JB):**	2.59e-09
Kurtosis:	4.676	**Cond. No.**	1.00

Notes:
[1] R^2 is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.

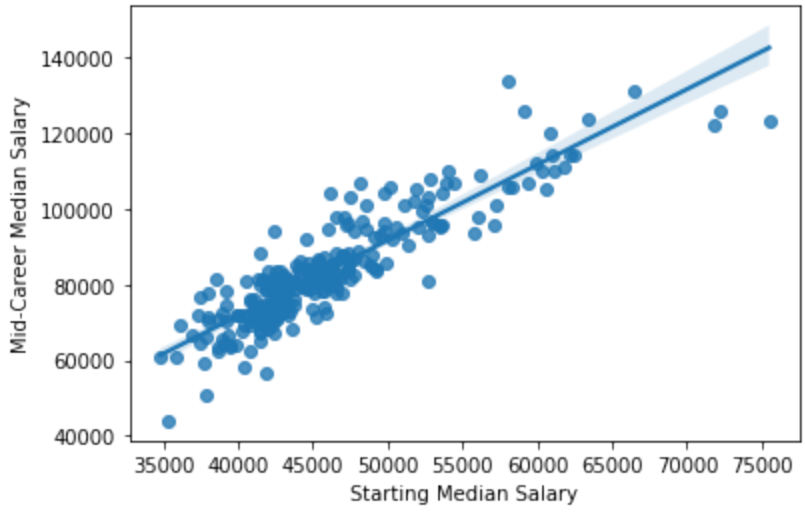
We can see that the R^2 value of .994 means that 99.4% of the variability observed for Mid-Career median salary can be explained by our model. Therefore, the data is close to the fitted line in the regression. The p-value for Starting Median Salary is 0.0. Therefore, we can reject the null hypothesis and conclude that the Starting Median Salary is a good predictor of the Mid-career median salary.

In [334…

```
seaborn.regplot(salary_college_df, x = 'Starting Median Salary', y = 'Mid-Career Median Salary')
```

Out[334]:

```
<AxesSubplot: xlabel='Starting Median Salary', ylabel='Mid-Career Median Salary'>
```



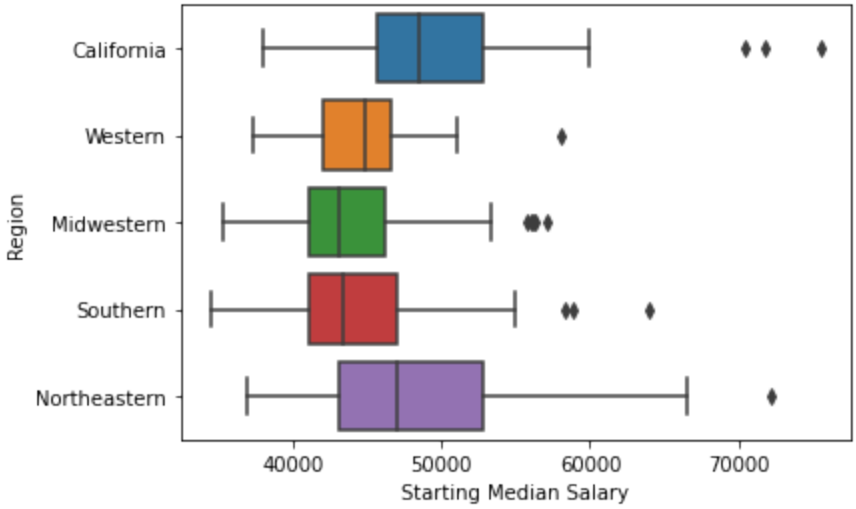
In order to visualize the regression, we plot a regplot to see that in general, as starting salary increases, mid-career salary increases. Therefore, we can conclude that starting salaries can predict mid-career salary and that if someone has a high starting salary, for example, their mid-career salary will generally be higher than someone with a lower starting salary. Salaries also generally increase over time. This makes sense as people tend to get raises and get better jobs further in their career.

Other Analysis

What is the best combination of major, region, and college that will yield the highest paying starting and mid career salary?

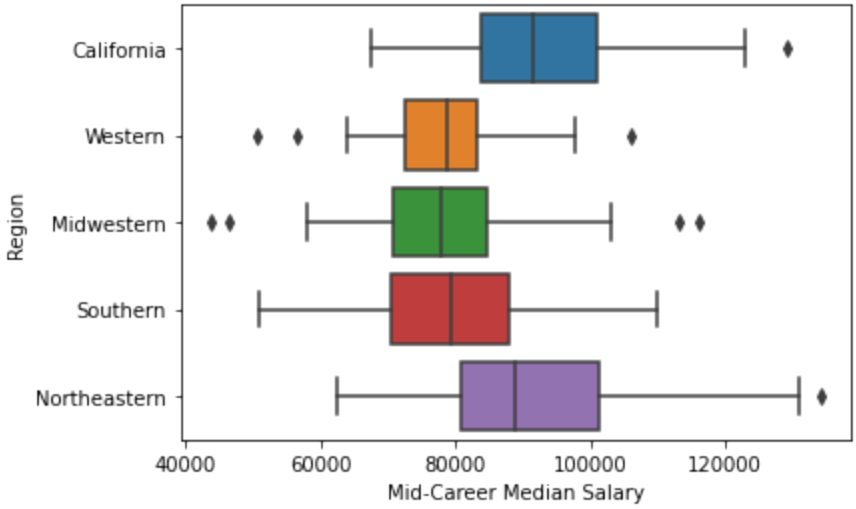
In [335…
```
college_df = salary_college_df[['School Name', 'Starting Median Salary', 'Mid-Career Median Sala
region_df = salary_region_df[['School Name', 'Region', 'Starting Median Salary', 'Mid-Career Med

seaborn.boxplot(data=region_df, x='Starting Median Salary', y='Region')
```

Out[335]:
```
<AxesSubplot: xlabel='Starting Median Salary', ylabel='Region'>
```



In [336…
```
seaborn.boxplot(data=region_df, x='Mid-Career Median Salary', y='Region')
```

Out[336]:
```
<AxesSubplot: xlabel='Mid-Career Median Salary', ylabel='Region'>
```

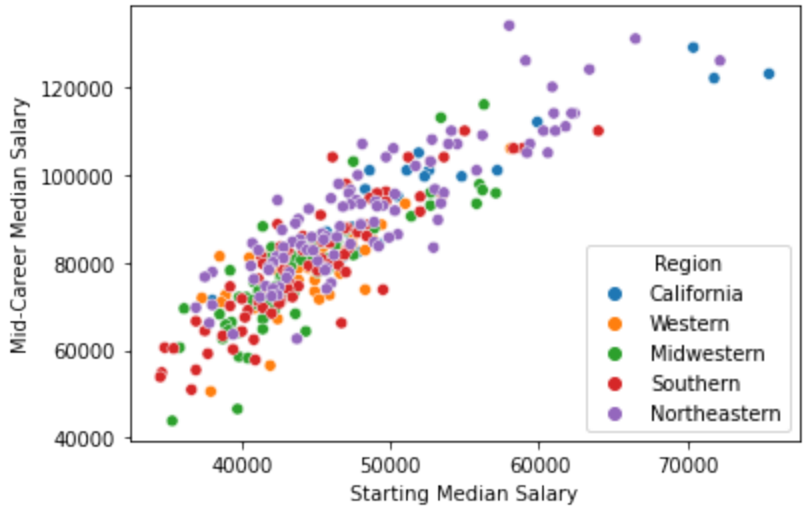


By plotting the box and whisker plots for Starting and Mid-Career Salaries against regions, we can see that California and Northeastern regions tend to have the highest salaries for both Starting and Mid-Career.

Therefore, we can predict that attending school in one of these regions will result in the highest paying salary.

In [337…
```
seaborn.scatterplot(region_df, x='Starting Median Salary', y='Mid-Career Median Salary', hue='Re
```

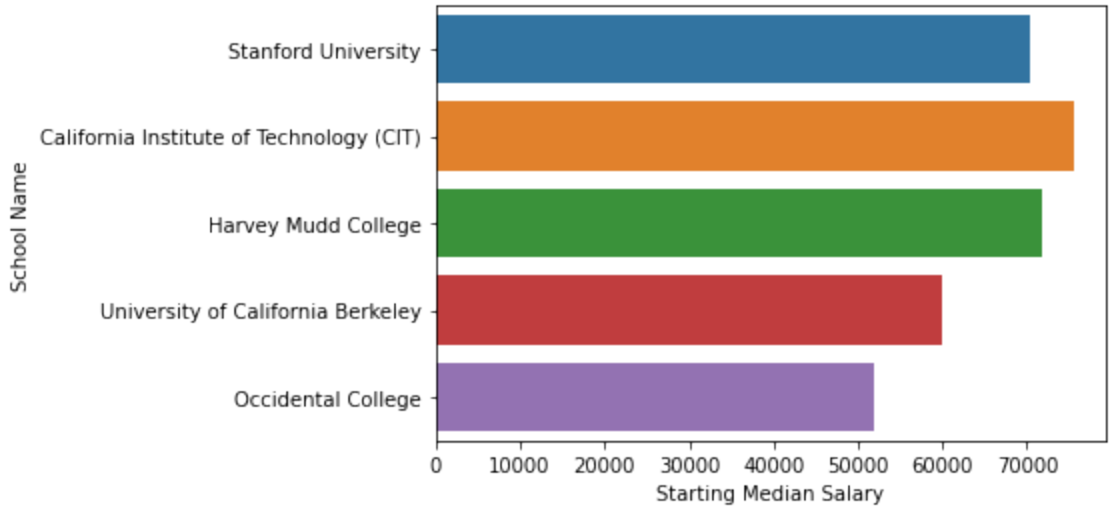
Out[337]:
```
<AxesSubplot: xlabel='Starting Median Salary', ylabel='Mid-Career Median Salary'>
```



From this scatterplot, we can see that the highest paying Starting Median Salary is in California and the highest paying Mid-Career Median Salary is in Northeastern.

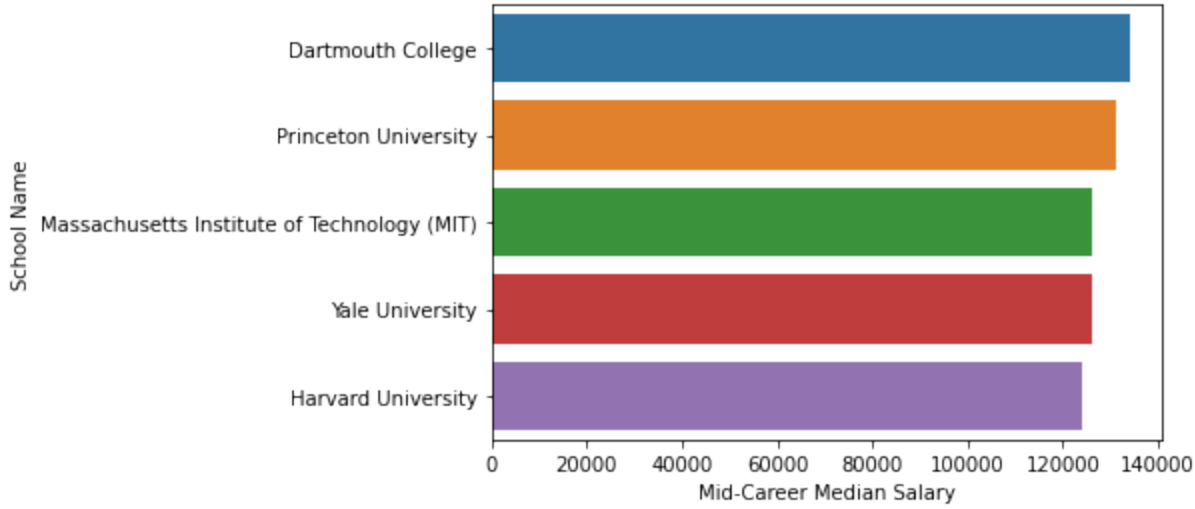
In [338…
```
california_df = region_df[region_df['Region'] == 'California']
california_df.sort_values(by = 'Starting Median Salary', ascending = False)
seaborn.barplot(california_df[:5], x='Starting Median Salary', y='School Name')
```

Out[338]:
```
<AxesSubplot: xlabel='Starting Median Salary', ylabel='School Name'>
```



In [339…
```
northeastern_df = region_df[region_df['Region'] == 'Northeastern']
northeastern_df.sort_values(by = 'Mid-Career Median Salary', ascending = False)
seaborn.barplot(northeastern_df[:5], x='Mid-Career Median Salary', y='School Name')
```

Out[339]:
```
<AxesSubplot: xlabel='Mid-Career Median Salary', ylabel='School Name'>
```



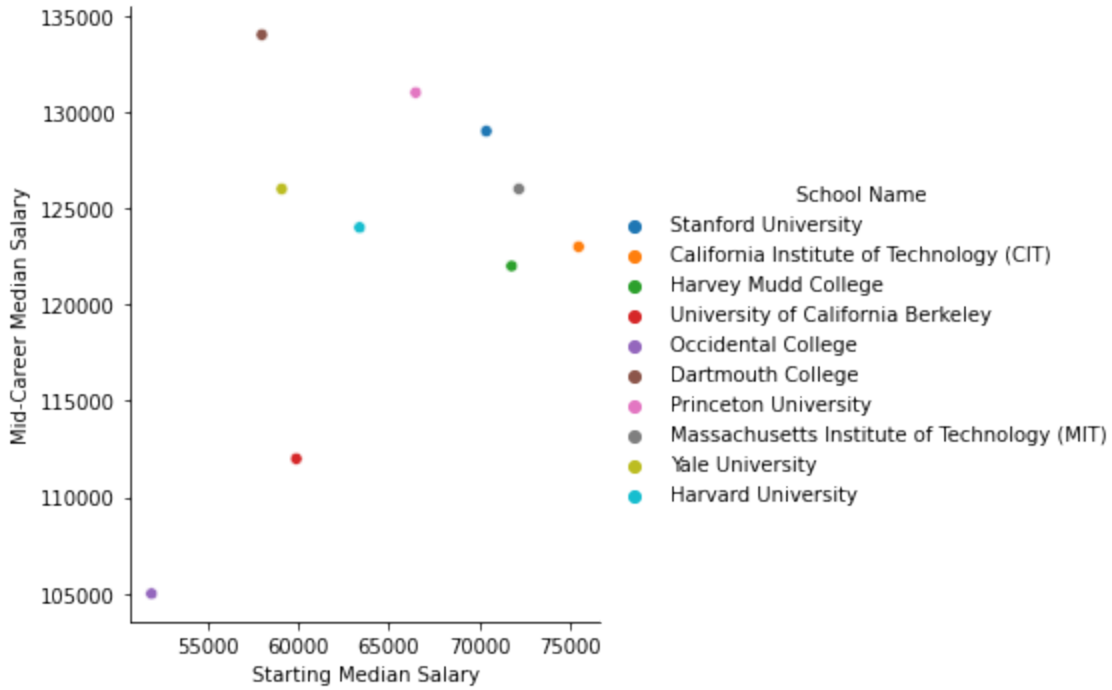
Therefore, we can conclude that the highest paying Median Salary has resulted from California Institute of Technology (CIT) and the highest paying starting salary has resulted from Dartmouth College.

In [340…

```
top_start = list(california_df[:5]['School Name'])
top_mid = list(northeastern_df[:5]['School Name'])

top_sal = top_start+top_mid

region_df.reset_index
top_sal_df = region_df[region_df['School Name'].isin(top_sal)]
seaborn.relplot(top_sal_df, x='Starting Median Salary', y='Mid-Career Median Salary', hue='Schoo
```

Out[340]:

```
<seaborn.axisgrid.FacetGrid at 0x7fddd4895b20>
```

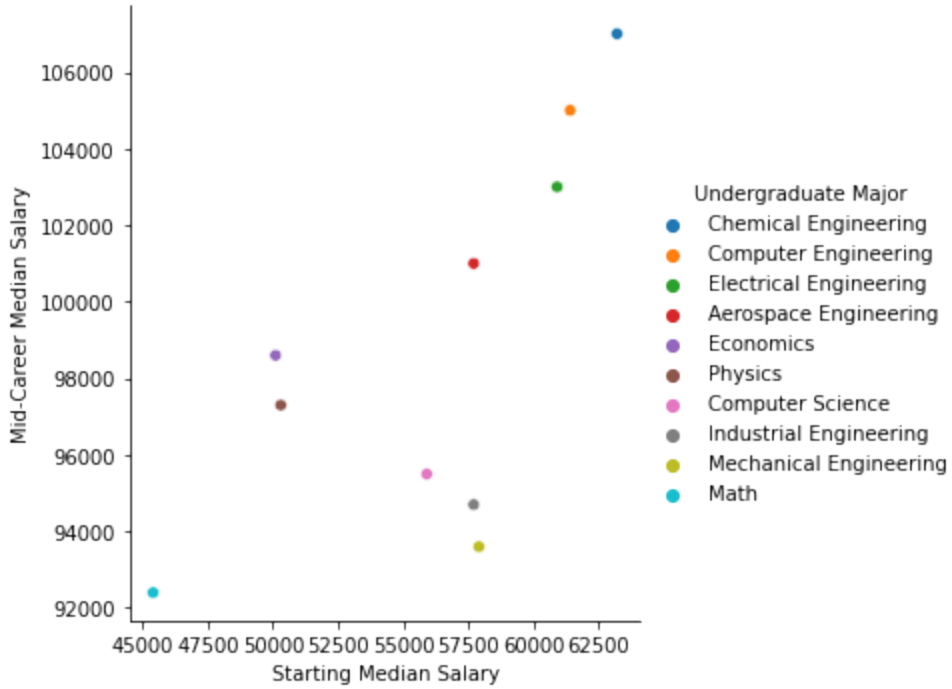


Based on this plot, we can see that even though CIT has the highest starting median salary and Dartmouth has the highest mid-career salary, they have lower mid-career and starting salaries, respectively. Therefore, we can conclude that Stanford University, Princeton University, and MIT has a better combination of starting and mid career salaries.

In [341…

```
degree_mid = degree_df.sort_values(by=['Mid-Career Median Salary'],ascending=False)
seaborn.relplot(degree_mid[:10], x='Starting Median Salary', y='Mid-Career Median Salary', hue='
```

Out[341]: <seaborn.axisgrid.FacetGrid at 0x7fddd48b0550>



We can see that the highest paying major for both Starting and Mid-career salary is Chemical Engineering. Therefore, we can conclude that the best combination for earning the highest paying Starting salary is majoring in Chemical engineering at California Institute of Technology in California. The best combination for earning the highest mid-career salary is majoring in Chemical engineering at Dartmouth in the Northeast.

These conclusions have been based on the data that we are given, however, there are some factors that are not taken into account. For example, the degree data does not include the school that the degrees are earned from and the schools resulting in the highest salaries could be earned by people that do not major in the highest paying majors concluded from the degrees dataset.

Interpretation and Conclusions

Exploring the data and running tests on Cornell postgraduate and National postgraduate degree salaries, we have concluded that the median starting salaries are greater than national postgraduate median salaries for more than 95% of majors. First, we visualized the data with a boxplot and histogram to determine that Cornell salaries are higher overall than national salaries. Then, we plotted a graph to look at each specific major for Cornell and schools nationally. Therefore, we were able to determine which majors resulted in higher salaries. We can observe that Journalism, Art History, and Music, for example, are some of the few majors with higher national postgraduate salaries than Cornell. Finally, performing a T-test and Mann-Whitney U test with resulting p-values close to 0 led us to reject the null hypothesis and conclude that overall, Cornell starting salaries are higher than starting salaries at other schools with equivalent majors. It is important to note that all these majors are not exactly equal, as Cornell has majors that differ from the national majors given.

For our second hypothesis, which looks into whether STEM majors make more in starting and median salaries than Liberal Arts majors, we first visualized the top majors for starting and mid-career salaries using bar graphs, which appearing to be mostly STEM majors. Then, we performed logistic regression to determine how accurately knowing starting and mid-career salaries could predict whether someone majored

in STEM or not. For each $1,000 increase in starting median salary, the odds of the major being STEM are multiplied by about 1.334. On the other hand, for each $1,000 increase in mid-career median salary, the odds of the major being STEM are multiplied by about 0.969. We can conclude that STEM majors tend to have higher starting salaries than non-STEM majors right out of school. However, mid-career salaries cannot be predicted to be higher as much for STEM majors because the salaries tend to be more similar to a non-STEM major once they are at mid-career.

Our third analysis verified our hypothesis that Ivy League graduates have a greater increase in salary compared to other types of schools. First, we used scatter plots to determine that Ivy League and Engineering schools have the highest paying Starting and Mid-Career salaries. Then, running a Naive Bayes model with a score of .56 and developing clusters for each school type, it is not obvious which school type can best predict starting and mid-career salaries. Therefore, we change the data to cluster Ivy League schools and Engineering schools and Ivy League and Non Ivy League schools in order to better examine our hypothesis. The score of .88 for the Ivy League, Engineering cluster and the other schools' cluster from the Naive Bayes model informs us that Ivy League and Engineering schools are a better predictor of higher starting and mid-career salaries. The final model, comparing Ivy League and non Ivy League yields the best score of .96 for starting and mid-career salaries combined and for mid-career salaries. Therefore, we can conclude that attending an Ivy League schools will increase chances of earning a higher salary and that both high starting salaries and high mid-career salaries are good predictors of attending an Ivy League school.

For our fourth hypothesis, we were able to explore how salaries change over time. To visualize this, we first plotted a histogram displaying starting and mid-career salaries. This allowed us to see that salaries generally increase over time, but the mid-career salaries are more spread out than the starting salaries. Then, we ran a linear regression model with coefficient 1.99 and score .80 in order to see that starting salary is a decent predictor for mid-career salary. As starting salaries increase by a factor of 1, mid-career salaries increase by a factor of 2. So, we can conclude that having a high starting will predict having a higher mid-career salary.

Finally, we explored the question as to what is the best combination of major, region, and degree that will yield the highest starting and mid-career salary. With some exploratory analysis, we conclude that Northeastern and California regions have the highest salaries, for both starting and mid-career. Focusing in on the schools in these regions with the highest salaries, we explore which of these schools has the highest starting and mid-career salaries. CIT has the highest starting salary and Darthmouth has the highest mid-career salary, but we can see that there are other schools, such as Standard, with high starting and mid-career salaries. The highest paying degrees are engineering degrees, with Chemical Engineering being the highest paid. We can provide the conclusion that majoring in Chemical Engineering at one of these schools will yield the highest starting and/or mid career salary. However, since the degree dataset represents the highest paying majors overall, we don't know which schools are best to earn a high paying Chemical Engineering degree. Based on the data we have, can predict that a degree Chemical Engineering at a school in the Northeast or California, such as Stanford, CIT, or Dartmouth, will yield the highest paying salary at the time that this data was collected.

Limitations

There are some NaN values in the Salaries by Region and Salaries by College, but these NaN values are only in the 10th and 90th percentiles. We have complete data for median and 25th and 75th percentile data. We also found that the data on Salaries by Major is complete with no NaN values.

One limitation was that the Cornell majors did not fully correspond to national majors, as Cornell has very unique majors. We made our best assumptions of corresponding majors (i.e. Operations Research and Information Engineering and Industrial Engineering, Electrical and Computer Engineering and Electrical Engineering, etc.) and had remove some majors at Cornell that were not included in the national dataset along with some majors in the national data which didn't correspond to a major at Cornell. We decided to neglect these majors from our analysis and focus on majors that did match up for our comparison analysis. This limitation impacts our findings because our conclusion of whether or not Cornell salaries are significantly higher than national averages will not apply to Cornell majors such as Policy Analysis and Management (PAM) and Environment & Sustainability (among others) which are both very popular majors at this university. This is an unfortunate result of national data not being available for majors of these categories. In addition, not including majors that are not offered at Cornell enforces us to alter our hypothesis to see if Cornell Postgraduate salaries are higher than national averages for majors that are offered at Cornell and have national data available.

When collecting the salary data by major from the Cornell postgraduate outcomes, we initially chose to have the years range from 2016-2018 because that data described the salary data from approximately 5 years ago. We decided to use this range because we were looking for majors that correspond with the majors in the WSJ national data, which was last updated 5 years ago. However, there were certain majors that did not have sufficient data. So, we used data from a larger range of years to account for the missing data. For example, anthropology did not have enough responses to show sufficient data for salary statistics. So, we had to use the range of 2016-2021 to get enough data. We realize that having inconsistencies in data collection is not good practice and we would not have resorted to doing so if we had complete data on all majors at Cornell. However, we thought that the benefits of not including majors outweighed the negatives of using data from more recent years. Using data from more recent years slightly impacts the validity of our findings because our model does not take into account the impacts that inflation would have had on salaries from 2018-2021, and directly compares it with the WSJ data from 2017 (about 5 years ago).

Source Code

Github Repository Link: https://github.com/juliavanputte7/2950

- Contains all datasets
- Contains project notebook with code

Acknowledgments

For parts of our data analysis, we were inspired by some projects we found on Kaggle that utilized the same National Salaries dataset that we used in throughout our analysis.

Specifically in this project (https://www.kaggle.com/code/skalskip/what-to-expect-after-graduation-visualization), visualization "3.3. Time makes all the difference" was used to help us complete our first visualization in our Hypothesis 4 section.

In addition, "Graph 2: Starting & Mid Career Median Salary by Major" from this other project (https://www.kaggle.com/code/cdelany7/exploration-of-college-salaries-by-major) was used to help us form the last two visualizations in the Hypothesis 1 section.

The code below for for creating a Bayesian plot was sourced from the link below and edited to fit our data. source = https://jakevdp.github.io/PythonDataScienceHandbook/06.00-figure-code.html#Gaussian-Naive-Bayes

These were the only other places we directly referenced when writing code for our project, other than various StackOverflow posts that I believe would would be excessive to include each and every one.

We would also like to acknowledge the 2950 Professors that helped us in learning this material that influenced our analysis.