Sydney Lauer, Lauren Belous, Maddie Kelsch

SI 330

April 20 2023

Final Project Report

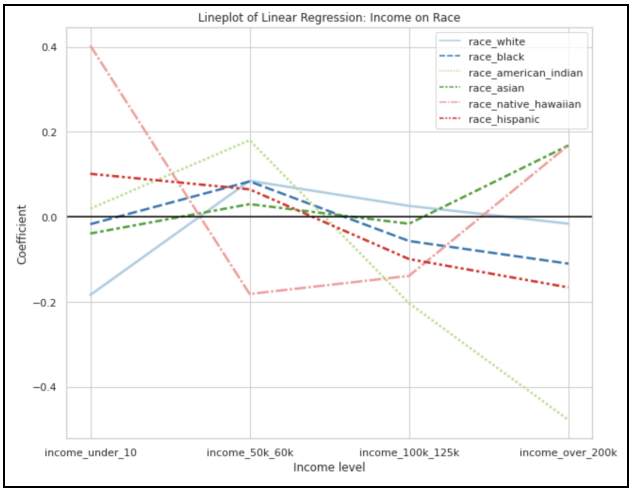
Result #1: How do Racial Demographics Affect Income Levels?

Skill: Linear regression analysis involving more than 5 independent variables

We investigated whether there is a relationship between racial demographics and income levels within Detroit census tracts. The question seeks to investigate whether the racial makeup of each tract has any correlation with the makeup of the various income levels within each tract.

This question matters because racial biases and stereotypes can impact the career trajectories of minority racial groups, preventing them from getting hired at new jobs or receiving promotions and raises at their current jobs. By investigating whether the presence of certain racial groups consistently corresponds to the presence of certain income levels within Detroit, we can find insights into whether these racial groups are likely facing racial discrimination relating to their careers. If there is evidence of racial discrimination, it is then important to investigate the underlying cause and how it can be remedied, perhaps with increased education focusing on DEI or even with new legislation created by Detroit lawmakers.

We approached answering this question by running linear regressions on various income levels in relation to the percentages of the 6 racial groups present in each census tract. We first had to merge the demographic data frame that contained the data about the racial groups with the income data frame containing the data about the income levels. We then ran regressions for four different income levels that represented the lowest income level, the lower quartile income level, the upper quartile income level, and the highest income level. Lastly, we plotted the correlation coefficients using a line plot to help us visualize the data and identify any trends.



Based on the plot, there does not seem to be a correlation between race and income. If there was a strong correlation between the two categories, we would expect to see more strictly linear trends where there was a higher correlation coefficient for income under 10k and increasingly lower coefficients all the way to income over 200k, or vice versa. The only racial group that this linear trend held true for was the Hispanic population, which is more closely associated with lower income levels and less frequently associated with higher income levels. This could indicate a potential racial bias against the Hispanic population within areas of Detroit. However, by looking at other racial groups, such as the Native Hawaiian group that has a high correlation with both the income under 10k category and the income over 200k category, we can answer that there is not a definitive association between race and income.

In the future, we would like to run regressions for more than four income levels to get a more accurate picture of the relationship between income and race. We would also like to run regressions with other independent variables in addition to race such as age and gender to see if those had a strong impact on the income levels within each census tract.

Result #2: What Types of 911 Calls Occur Most Frequently?

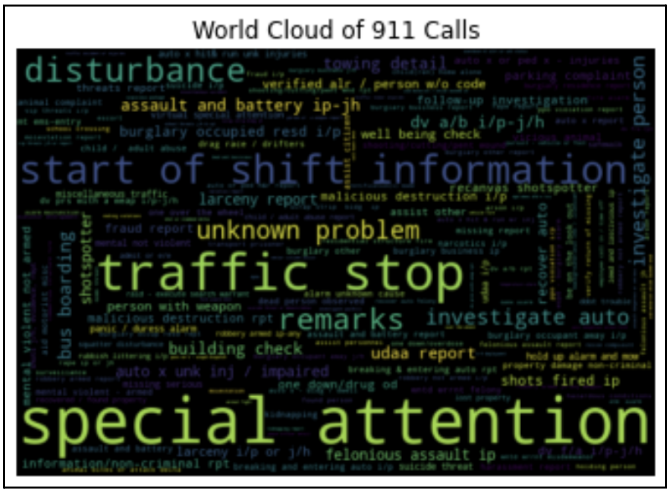
Skill: Text analysis requiring comparing word frequencies

We investigated what kinds of calls to 911 occur most frequently within Detroit. This question is designed to provide insight into what types of crime occur most frequently within Detroit, although it can truly only answer what kind of crime is reported most frequently, as some crime takes place within Detroit without being reported.

This question matters because it provides some insight into what type of crime occurs most frequently and what Detroit citizens' greatest concerns and issues are. Knowing this information would be extremely helpful for legislators, who can draft new legislation in response to crime that occurs extremely frequently in an attempt to lower the rate at which those illegal acts occur. This information would also be helpful to law enforcement to encourage them to

make an active effort to take proactive measures against those types of crime and prevent them from happening in the first place.

We approached answering this question by performing text analysis utilizing word frequencies. We kept track of the frequencies for each type of 911 call included in the database using a counter. The counter of the frequencies allowed us to create a word cloud to help us easily visualize which types of phone calls occurred the most frequently.



The word cloud uses text size to clearly indicate that “special attention” calls occur most frequently. A “special attention” call is a “proactive” type of call that is “usually generated at the request of the public for the police to focus on a particular area or address” (“Manchester Police Public Data Dashboard.”). This indicates that it is important to Detroit citizens to play an active role in preventing crime before it occurs, and also positively indicates that more frequently are calls taking place before a crime has been committed rather than after. Traffic stops are also a frequent call, which indicates that perhaps Detroit citizens could use better education when it comes to road rules and driving etiquette.

This information is extremely interesting when analyzed in conjunction with a report by *The Detroit News* that found Detroit to have a crime rate of 2,248.44 violent crimes per 100,000 residents in 2020, the second highest amongst big cities in the US (Harding and Hunter). These crimes include assault, robbery, rape and criminal homicide, none of which can be clearly identified in the word cloud. The cause for this is likely that as frequently as these crimes occur, they still occur less frequently than some of the more banal calls that Detroit police receive pictured in the word cloud, or that these crimes are not reported over the phone.

In order to dig deeper into this question, we would like to compare the information presented in the 911 calls database with crime reports that perhaps did not take place over the phone but through other means, such as online or in-person at police stations, or gain access to information detailing arrests and tickets throughout Detroit. We believe this would provide a better understanding of what types of crime take place in Detroit because not all crimes are reported over the phone.

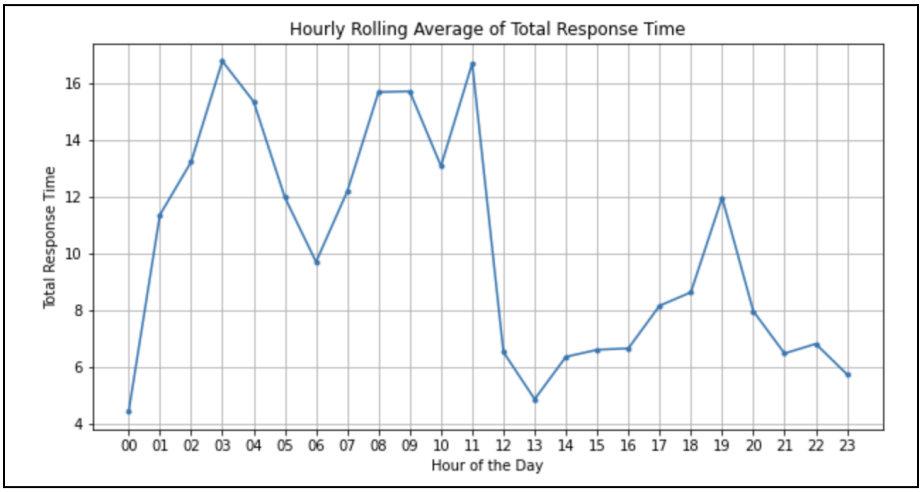
Result #3: How do Police Response Times to 911 Calls Vary Throughout the Day?

Skill: Time series analysis requiring rolling average and groupby

We investigated whether the total response time to 911 calls varies based on the time of the day. We also wanted to determine at which times police were the fastest and slowest in their response times.

This question matters because it can provide insight as to whether the Detroit Police need to redistribute how many police officers are working at white times. If there are times during the day in which response times take significantly longer, then there should be more police officers working at those times. This matter is serious because the time at which the police arrive on the scene of a crime could mean the difference in whether a person lives or dies or whether a crime is committed or not committed.

To answer this question, we performed a time series analysis using a rolling average and group by. We grouped by the hour of the day and used a rolling average with every 5 entries so as to not let occasional fluctuations influence our results and focus more on long-term trends. We chose to plot our data using a line plot to understand how the times fluctuate over the course of a day.



Our results tell us that average response times do vary throughout the day and seem to fluctuate without clear reason. Police respond the fastest, on average, in the afternoon from 12pm - 5pm and in the evening around 9pm - 12am. The police have the longest response times around 3 am in the morning and between 8 and 11 am. The sharp drop from 11am - 12 pm might indicate that more police officers start working midday or that fewer calls take place in the middle of the daylight. In the same vein, the climb from 12 am to 3 am might indicate police officers being allowed to go home for the night, or might reflect that more crime and thus more calls take place at night when less people are around and it is more difficult to see crime taking place.

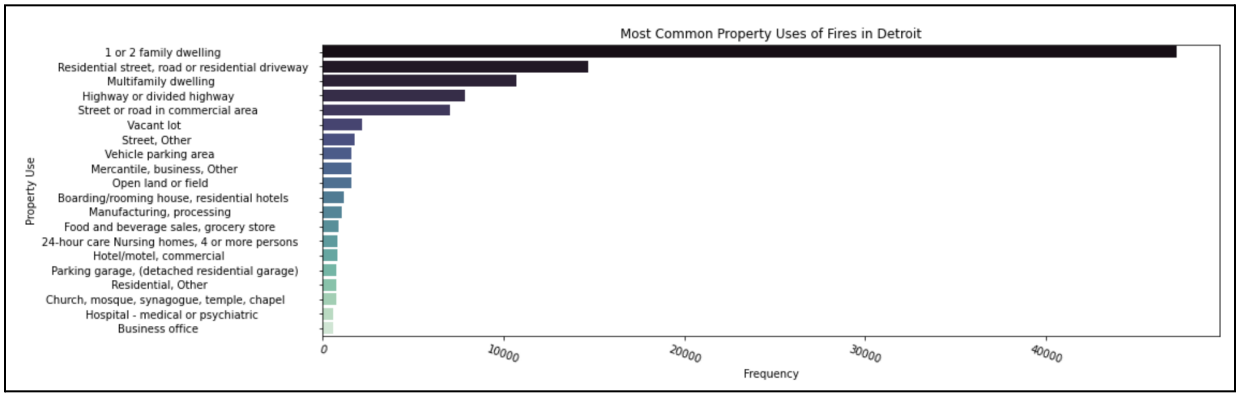
To dig deeper, we would like to take a look at the response time in relation to the amount of calls received at each hour, because it is quite possible that the reason there are faster or slower response times at certain times of the day is because the police receive less or more calls at those times.

Result #4: At What Type of Locations do Detroit Fires Occur Most Frequently?

We investigated the types of locations at which fires occur most frequently within Detroit. This question involves looking into the use of the properties at which fires have been reported according to the data in the database.

This question matters because if there are certain types of locations at which fires occur way more frequently, then it is clear those types of properties do not have the proper preventative measures in place and need to be redesigned in a way that helps them better prevent fires. This information is important for firefighters as well because they can pay special attention to those types of properties and can make sure there are fire stations located close by these properties. An additional solution would be making sure the owners of these properties know what to do in the case of a fire.

We answered this question by grouping data entries by their property use and using a counter to keep track of how frequently each property use occurred in the database. In order to easily process which types of properties occur the most frequently and how frequently they occurred in relation to properties, we decided to display our data using a barplot.



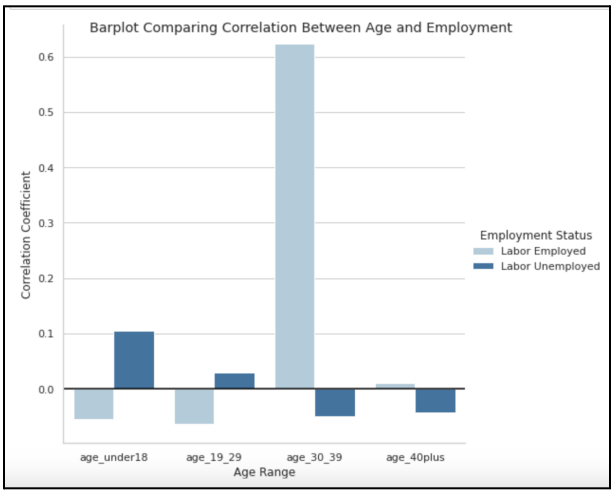
The bar plot indicates the fires in Detroit take place most frequently at “1 or 2 family dwellings”. It also indicates that fires take place at these locations much more frequently than any other type of location, with over 40,000 occurrences in the data frame and the second most frequent location occurring less than 20,000 times. This information is extremely valuable because it indicates that families in Detroit could benefit from more education regarding fire safety and potentially also better fire-related equipment such as smoke alarms and fire extinguishers. Interestingly enough, outside fires occur relatively frequently as well, as 3 of the most frequent property uses involve streets, roads, driveways and highways. The results indicate overall that residential areas and buildings are at the highest risks for the occurrence of fires.

Result #5: How does Age Distribution Affect Employment Levels?

This question asks us to investigate whether a correlation between the employment and unemployment levels within each census tract and the percentages of each age group within each census tract exists. In other words, we wish to discover if having a higher percentage of a certain age group or groups corresponds to higher employment or unemployment levels.

The answer to this question is important because it can reveal whether there is a problem of ageism when it comes to hiring new employees in Detroit and if job applicants that are much older or younger than other applicants are discriminated against and not hired simply because of their age. If so, it is important that measures be put in place to prevent against this kind of discrimination. Additionally, it is important to keep track of the predominant age ranges that make up the current workforce because an older workforce might indicate an incoming labor shortage in the future, in which case we would want to take active measures to encourage more young people to join the workforce.

We approached answering this question by first merging the demographic data frame that contained the data about the age groups within each tract with the work data frame containing information about the employment and unemployment levels. Next, we ran two linear regressions, one for employment levels against the 4 age groups and one for unemployment levels against the 4 age groups. Lastly, we took the correlation coefficients from the linear regressions and plotted them using a grouped barplot, which allows us to easily compare how the employment and unemployment correlations differ for each age group.



As we can see in the grouped bar plot, the highest correlation is between the employment level and the 30-39 age group, meaning that having a higher percentage of individuals aged 30-39 makes it more likely to have a higher employment level. This result is unsurprising, as individuals aged 30-39 are more likely to have families they need to provide for, own homes they need to pay rent on, etc. and thus need to be employed, which is not the case for many individuals younger than 18 or full-time students who do not have time to be employed in the

18-24 range. We can also notice that having a higher percentage of individuals age 40 or older has a negative correlation with higher unemployment levels, which is interesting because many individuals retire after they turn sixty years old, and so we would expect there to be a positive correlation with unemployment and not a negative correlation. Lastly, it is worth noting the negative correlations with individuals younger than 29 and employment levels, perhaps indicating that younger populations have lower employment levels, which makes sense of reasons discussed above.

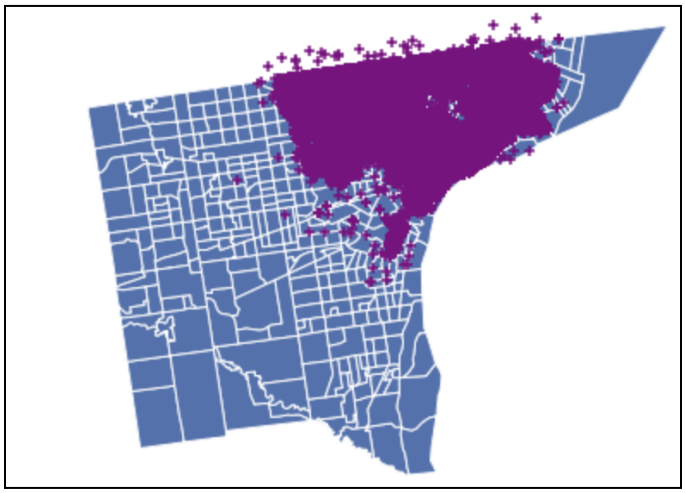
To dig deeper, we would like to look more closely at the age_40plus category to identify how many individuals are between 40 and 60 and how many are older than 60. Doing so would help clarify how retirement plays a role in the data and employment levels.

Result #6: Where within Detroit do calls to 911 occur most frequently?

This question asks us to investigate the areas of Detroit in which the greatest amount of phone calls to 911 take place. We are looking to see if there is one area or multiple areas in particular in which phone calls are concentrated, or if there is an even spread throughout the city.

This question matters because the amount of phone calls to 911 corresponds to how much crime is taking place in a given area. If phone calls are concentrated in certain areas in Detroit, then it is likely that crime occurs more frequently in those areas. If that is the case, then police could use this information to reallocate officers, stations, and other resources to those areas that need more attention.

We approached answering this question by first plotting a map of Detroit broken into census tracts. We converted the data frame containing the information of 911 calls to a pandas dataframe in order to be able to plot where each call took place. We then plotted the points for each call on top of the Detroit census tracts map.



As we can see in the plot above, the phone calls are not spread out throughout Detroit but are concentrated in the northeast part of the city in what is considered District 3. The amount of

phone calls from this region likely indicates that crime occurs more frequently in this area than in other parts of Detroit. However, another reason for this spatial distribution of phone calls could be that there is a higher concentration of the population in these areas which could account for the increased number of phone calls without crime necessarily taking place in these areas more frequently.

The next steps we would take would be to zoom in on District 3 and just focus on that area of Detroit to see the distribution within that highly concentrated area, which might tell us more information about the types of neighborhoods that these phone calls are predominantly coming from.

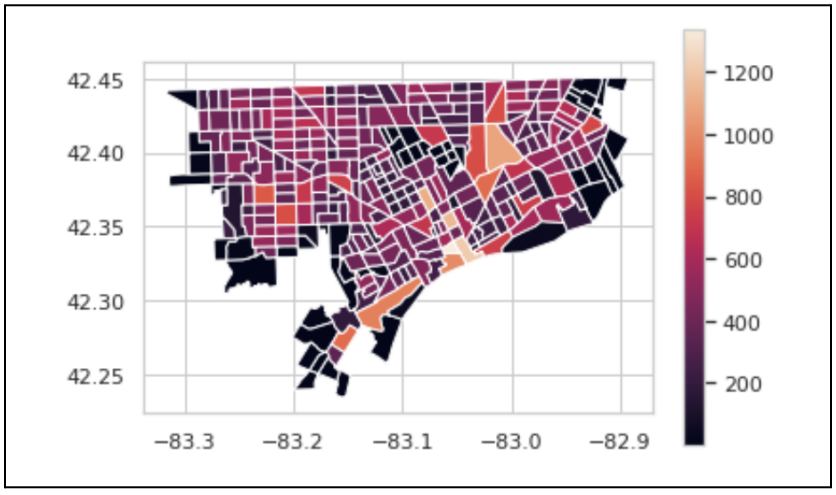
Result #7: Which areas of Detroit have the most fires on average?

Skill: Spatial analysis requiring local or global autocorrelation

This question asks us to investigate which parts of Detroit see the most frequent occurrence of fires. Specifically, we want to look at the various tracts within the city and see which tracts, on average, have the highest occurrence of reported fires.

This question matters because identifying which tracts have the most fires can allow the Detroit fire department to pay special attention to those tracts, potentially allocating extra resources to those areas when fires take place in those tracts. In addition to reactive measures, this information can help lead the city to identify why fires occur more frequently in these areas and how to take proactive measures to prevent so many fires from starting in these areas.

Our approach to answering this question began by converting the fires dataframe to a geopandas dataframe so that we could perform spatial analysis. We used a sjoin to merge this geopandas dataframe with the dataframe containing the information to plot the Detroit census tracts. With this new joined dataframe, we plotted the fire count for each census tract on the map of Detroit.



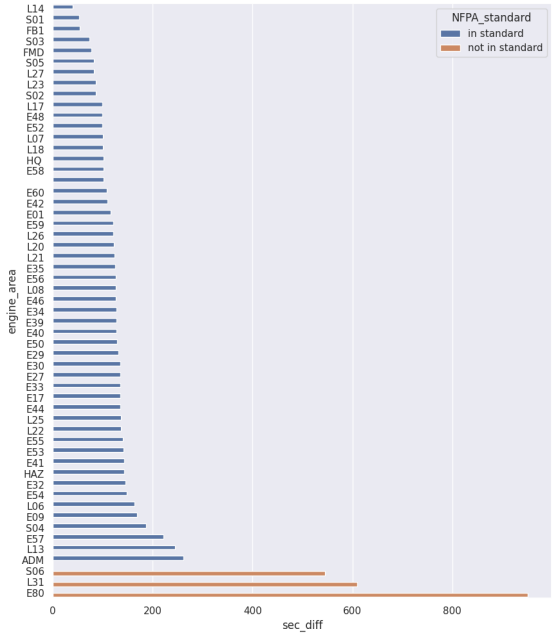
Interestingly enough, unlike the spatial plot for the most frequent 911 calls, the locations where fires occur most frequently are not concentrated in one area of Detroit. However, there does seem to be a pattern that on the outsides of the city, fires seem to occur less frequently, especially on the east and west sides of the city where the tracts average less than 200 reported fires. It is likely that the most of the population is concentrated in the center of the city, and since we know that fires take most frequently at residential homes and streets according to Result #4, then it makes sense that more fires would occur where the population is more concentrated and there are more residential areas in the center.

In order to dig deeper, we would want to look at common characteristics between the tracts with over 1000 reported fires to try to identify why these tracts have the highest number of fires on average.

Result #8 - What is the average time between call and dispatch for each engine area?

The question is asking us to find the average time difference between call time and dispatch time for each engine area. This will allow us to see how long it takes each engine area to jump into action after receiving an emergency call.

Answering this question is extremely important to see which Engine Areas perform the best and respond to an emergency call the quickest on average. This will also let us know if / which areas potentially need more training. Furthermore, the "NFPA Standard 1710 establishes a 320 second or 5 minutes and 20 seconds 'response time' goal". Answering this question will reveal which engine areas are in code.



The graph above represents which engine areas perform the quickest by having the shortest average response time from top to bottom. By observing the barchart I can conclude that

on average, Engine Areas beginning with the letter L, S, and F have better response times than Areas beginning with letters E or A, although the data has many outliers / variance. The bars shown in red, at the bottom, reveal that those Engine areas are not on average within NFPA Standard.

If I wanted to dig deeper into the answers the chart above reveals I would potentially use a different representation of the data. Another option would be to show this data spatially on a map so we could further recognize trends.

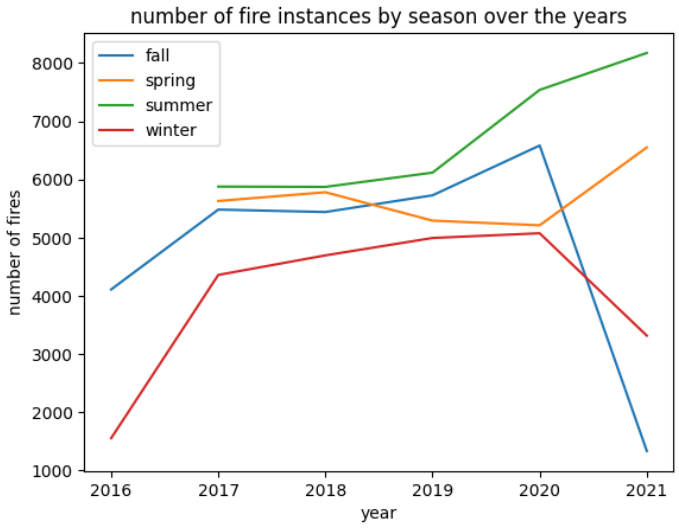
Result #9 - Are fires more likely to occur based on what season it is?

Skill: Time series analysis requiring using shift to compare different months or years on the axis

This question is asking us to analyze the amount of fires that occur over time and determine which season typically will have the highest numbers. We will understand these trends by examining how the average number of fires in a certain season change over the years included in the dataset (2016 to 2021).

This question matters because answering it allows us to notice potential patterns in global warming. This information may be very helpful to environmentalists in Michigan and around the nation. Answering this question can also allow us to draw other conclusions to better prepare fire districts in the area.

Our strategy for answering this question was to build unique functions that can be applied to this massive dataset to use datetime analysis and acquire the year and season of each fire instance. After this task was completed, we were able to use a groupby to organize the new columns into a useful new dataframe. This was then plotted to display the number of fire instances by season over the years with the number on the y-axis and year on the x-axis. The colorful lines display unique trends per season over the years.



The answer to this question shown from the plot is that in past years, the number of fire instances in Detroit Michigan, generally has increased for each season, as they all seem to move linearly. This reveals that each season from 2016 to 2020 has gotten relatively drier / hotter. However, I can't help but wonder what caused the drastic changes in all seasons from 2020 to 2021. But, I can assume that because of the record high number of fire instances of summer 2020-2021, there would be a significant change in the next occurring season being fall 2020-2021 causing the decrease there. After searching around on Google, many sources confirmed the spike in fires for the summer of 2021. The Detroit News wrote an article about the "130-acre wildfire" that took place during that season. The article also noted how dry the conditions were during this period of time.

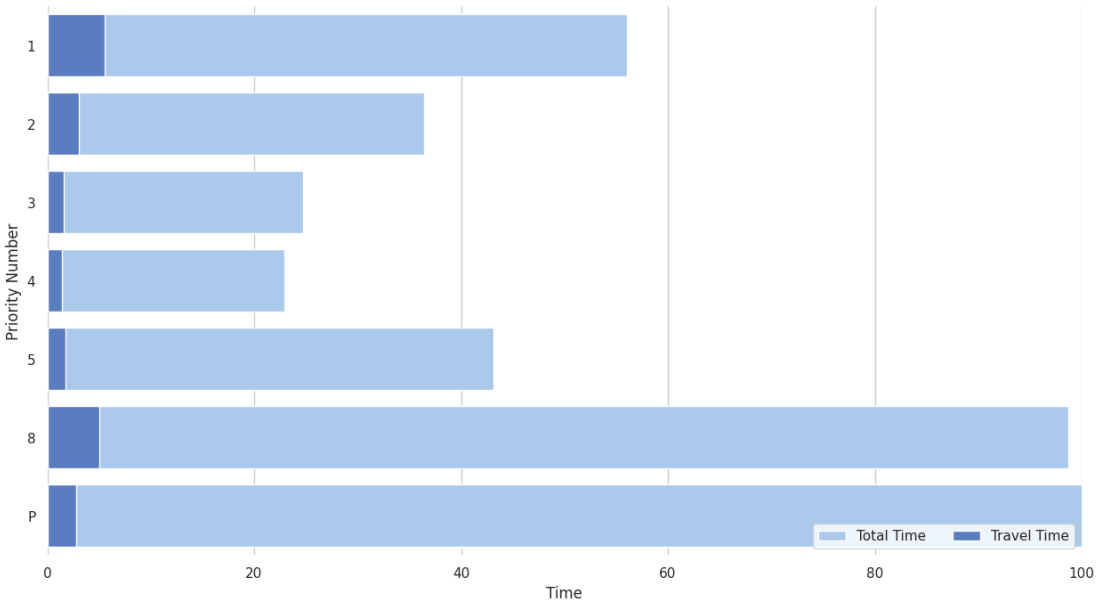
If I could do this question again I would probably use another graph to represent the data and see if that would reveal more trends. I think that plotting the information chronologically could potentially allow us to make more inferences about global warming and the impact it has.

Result #10 - How much of the total time does travel take up depending on call priority?

This question asks us to create a visual that represents how much of the total time is taken up by travel based on each call priority number. The priority number of the 911 call is based on how threatening to life an emergency is, 1 being the most urgent.

This question matters because it allows us to analyze a potential correlation between how long it takes to get to an emergency site in relation to the total time spent site based on how urgent the emergency is. Answers to this question may provide information to ER doctors and EMTs that can be useful to allocate resources. It also can help those answering 911 calls and connecting them to emergency vehicles such as ambulances more efficiently.

To answer this question I worked with the 911 calls dataframe and made two smaller data frames. The first smaller df was a groupby the priority number with the average travel time for each group. The second was also a groupby the priority number but with the average total travel time for each group. I then merged these two together on the priority column and plotted the remaining data. I chose a horizontal barchart using unique color codes to exemplify the differences.



The chart above does not necessarily reveal what I was expecting to see. For priority numbers 2-5 it seems that each average travel time takes up a similar portion of the total time bar. However, what I did expect was that for the most urgent emergencies (priority one) the travel time would take up a smaller portion of the total time bar as the emergency vehicles probably are moving very quickly to get to the site, and then proceeding to spend quite a bit of time there.

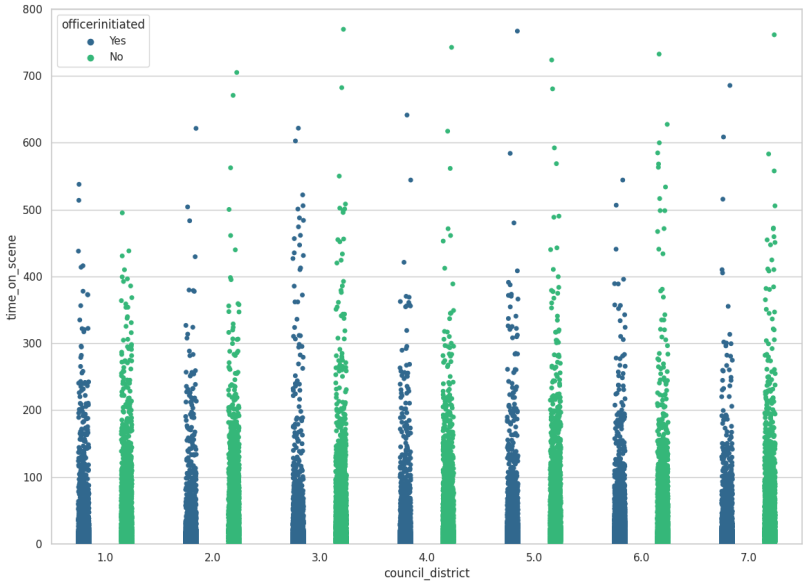
If I wanted to dig deeper into these results I would search for more data that would include how far away the location of the emergency site is from the vehicle's origin. Knowing this information would allow me to divide the amount of time it takes to get there by the distance away to see how fast the first responders are actually traveling. I think that this would prove to be more of a linear relationship in comparison to the priority number ranking.

Result #11 - Does the amount of time spent at a crime scene per council district change based on whether an officer is initiated or not?

This question asks us to recognize if there is a difference in the average amount of time spent at a crime scene by each council district change if an officer is initiated or not. Officer Initiated calls are where the officer rolls up on something.

This matters for the Michigan State Police (MSP) to understand how its Detroit council districts are performing. This also may give them insight into whether more officers should be in the field if it reduces the average amount of time that is spent at a crime scene. Understanding these correlations can allow the police department to better work with incoming 911 calls and allocate resources more efficiently.

I approached this question by parsing through the massic 911 calls dataset to only collect vital information to answer this question. These columns include time_on_scene, officerinitiated, and council_district. Because officerinitiated provides binary results, it worked perfectly as the hue of the plot. I chose to build a seaborn stripplot because it best represents the three categories of data.



The graph below displays some interesting results. Overall, I am very pleased to see the consistency between the council districts in Detroit. It seems that they all spend a similar amount of time at the scene of the crime on average whether or not an officer initiated the crime. However, in general it looks like there is more variability towards spending more time at the scene when an officer is NOT initiated across the board of council districts. This makes sense because without an officer present there may be more paperwork to fill out causing the time to extend.

If I wanted to dig deeper, my next step would be to include other councils outside of the one provided, maybe in another state. It would be interesting to see how these trends change when observing a different sample of data.

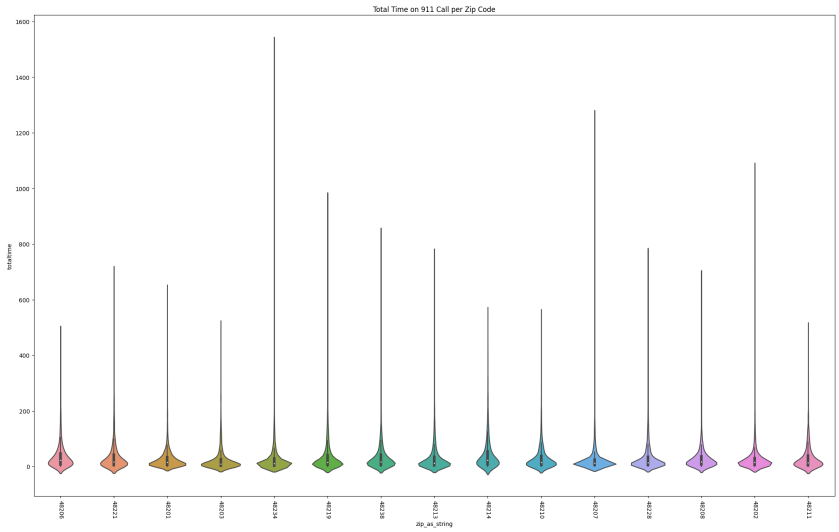
Result #12 - Does the total time of a 911 call change depending on the zip code it was in?

This question asks us to examine how zip codes affect the distribution of total time spent on a 911 emergency call. It wants us to find the statistical results of each zipcode to see if there are any significant trends.

This question matters because it helps first responders notice patterns in the trends of 911 calls in Detroit. Understanding the statistics of how much time is spent on a call, can be helpful in predicting future calls. This information can allow higher ups to better allocated resources and efforts across call precincts and distribute individuals across these zip codes. Also, it can ensure

that enough resources are available to each zip code to make sure emergencies are fully and equally attended to.

I approached this question with the idea to build a violin plot to show the statistical analysis of the information for each zip code. This plot will also demonstrate any differences between the zip codes. To clean up the data from the original form, I had to create a new column to convert the zips to strings, and then filter out a subset of the groups using a mask. Because There were 49 zip codes in the original data, I didn't think it would suffice to show them all as the plot would quickly get overwhelmed. So I sliced out the first 15 and plotted the remaining information.



The graph above shows that there isn't any dramatic difference in how zip code affects the distribution of time on a call. However, the plot reveals other information. For example, I can see that the first zip code represented, 48206, seems to have pretty routine calls and relatively quick results. This is unlike some of the other zip codes that show more variation in spending more time on their calls. This may be for a few reasons. Maybe their emergencies take longer to handle, or maybe the phone operations are walking the individuals through steps for longer. The more bulging data such as zip code 48207, shows that more data is concentrated in that area.

If I could re-do this analysis, I would run it on the last 15 zip codes from the dataset. Changing the set of groups I am running the code on will allow me to see how / if the results change.

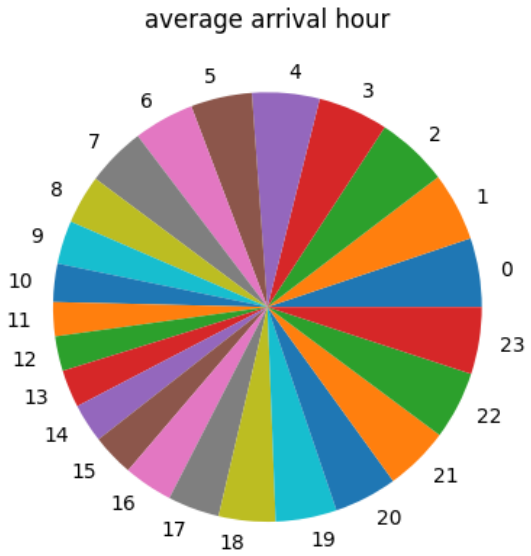
Result #13 - What are the common hours that firefighters arrive on scene?

This question asks us to see on average, what hours of the data are firefighters most often arriving on the scene.

This question matters because it will reveal answers to what hours of the day firefighters on average are most active and therefore most likely to be called to duty. Knowing this, can help

fire districts better allocate the times that their individuals are working and need to be on call. This can help the firefighters be ready at times that they are more likely to be needed. Also, it can reveal answers to when fires are most likely to occur which can provide answers to other trends.

To approach this question I had to convert the arrival times into datetime timestamps to be able to use the information to my benefit. After applying that function, I made sure to drop any NaN values. After that, I grouped by the arrival hour to count how many instances fall into each category. I chose to build a pie chart to represent the information I collected. This will allow us to easily see any differences in the proportions of the day that firefighters arrive on the scene of a fire.



The graph above surprised me in that the hours of the day that firefighters arrive on the scene of a fire are relatively evenly distributed. This is not what I expected as I assumed that more firefighters would be in action during the day when the sun is out. When it's sunny, land is probably drier which made me think that fires would be more likely to start. However, after examining the graph a bit more, I noticed that the opposite is actually true. If you look closely, the pie chart shows large slices for hours 22 ~ 3. In real time, these hours are 9pm to 3am.

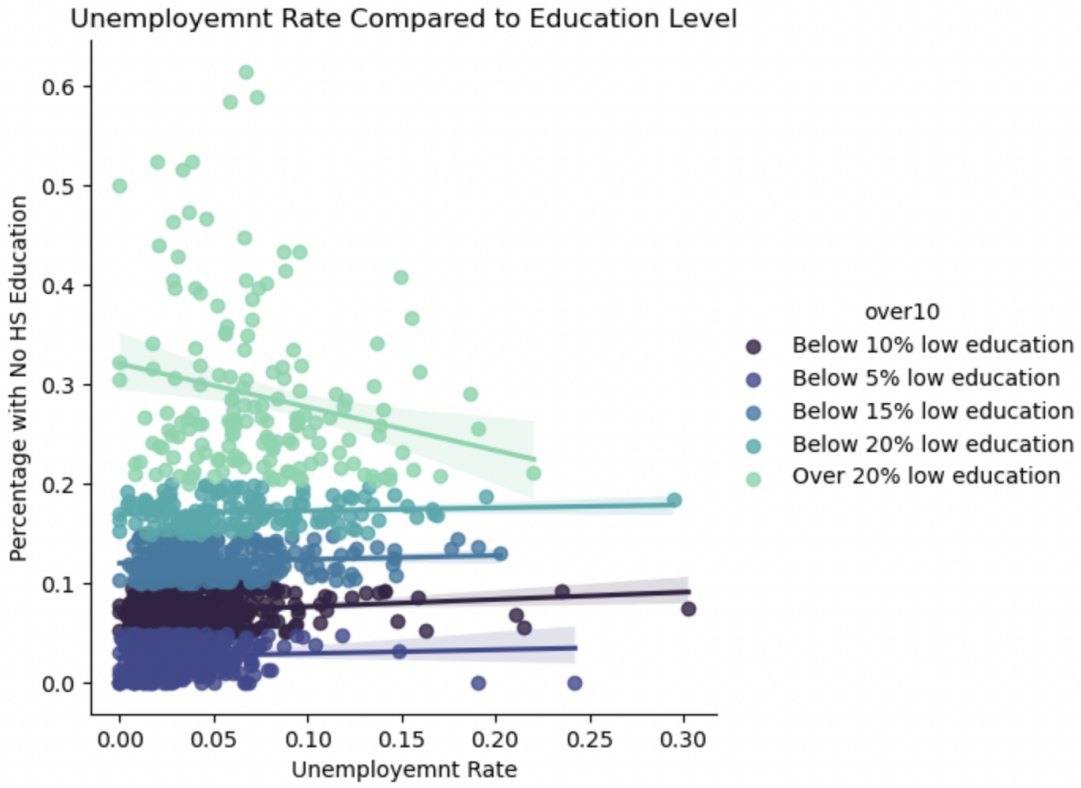
If I wanted to dive deeper into the answer to this question, I would include another characteristic to reveal more trends. For example, re-making this pie chart for every month of the year.

Result #14 - How does the level of education affect unemployment rates?

This question wants to find what impact overall education levels have on unemployment. This is useful to answer because it seems like a straightforward question but it is much more complex than people may realize.

There is a negative connotation that uneducated people are lazy and some may believe this would lead to higher unemployment. However, that may not statistically be the case (even if it is it doesn't mean people are lazy). By answering this question we can fight to either disprove that idea or give other reasoning if it is the case.

For this question, we merged two data sets and categorized education into percentages (intervals of 5 to 20%). This was done using a function that added up each row's no education proportion and no highschool proportion. We then created an lmplot at these different levels to indicate differences overall in varying levels of education and its effect on unemployment rates.



The plot shows that overall there is a positive correlation between unemployment and lower education levels. However, at different levels of education, there is almost no correlation (slope=0) between low education and unemployment. With that being said, lower education levels can be cited as a variable that impacts unemployment rates, as there was an overarching positive correlation. But at levels 1-4 (5%...20%) there isn't a correlation. Surprisingly, at the highest level of no education (20% plus) there is a strong negative correlation which could represent areas where there are more blue collar jobs that don't require traditional education.

Digging deeper into the data would require more data for what we would want to explore. The level of education breakdowns in the unemployed populations would be very interesting to

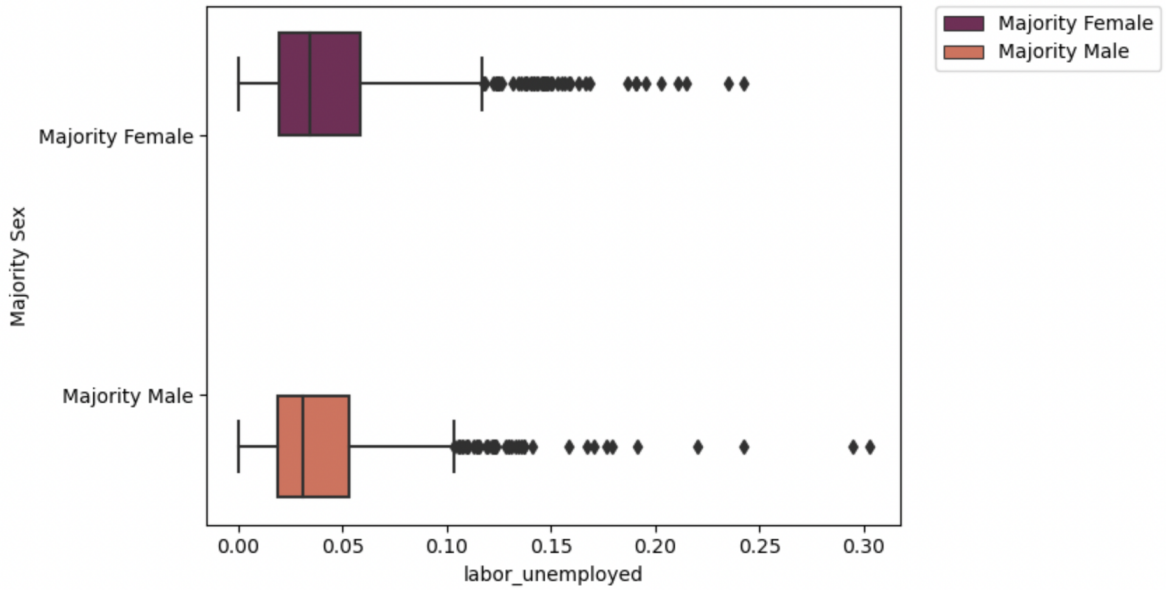
look at (i.e 50% of the unemployed population has a bachelors). With that data it would be much easier to address the true level of unemployment in each education level.

Result #15 - What is the unemployment rate like in more highly female concentrated areas?

With this question, we want to find if there is any relation between the percentage of females versus males overall and the unemployment rate.

We are answering this question to uncover if there is any implicit bias or historical bias that has led to varying unemployment rates based on gender. Women haven't been integrated in the work field for a significantly longer amount of time as their male counterparts (less than 100 years) and examining if this affects unemployment gives us insight to reoccurring ethical problems.

To answer this question, we merged the demographic and work dataframes from the census data. Then because there is no stat for each gender's unemployment rate, broke it down into majority male or female areas. From this, we plotted the distribution of unemployment in each respective majority gender section.



The boxplots show that unemployment levels in the different gender dominated areas are relatively similarly distributed. This would mean that gender isn't playing that large of a role in unemployment rates. However, the male distribution is notably smaller and has a lower max which does mirror slight inequalities. The outliers on the female boxplot are more concise and evenly together whereas the male outliers are more varied with a max unemployment rate of .3. This data was taken in 2021, the year after a recession, which makes us think the slightly unequal distributions are not by chance. The Covid-19 pandemic hit female workers the hardest, leaving

them with higher unemployment rates (Karageorge). This reasoning explains why all the points on the female box plot are larger than that of the males. Additionally, the .3 male data point could be from an area where there were a lot of male blue collar workers who long term lost their jobs. Considering the recent recession, the slightly disproportionate data makes sense.

We would like to look at education levels (like from the last question) along with gender and how all the different combinations affect unemployment distributions. It would be a lot of data to look at but, there may be unnoticed discrepancies because not all factors were considered in this analysis.

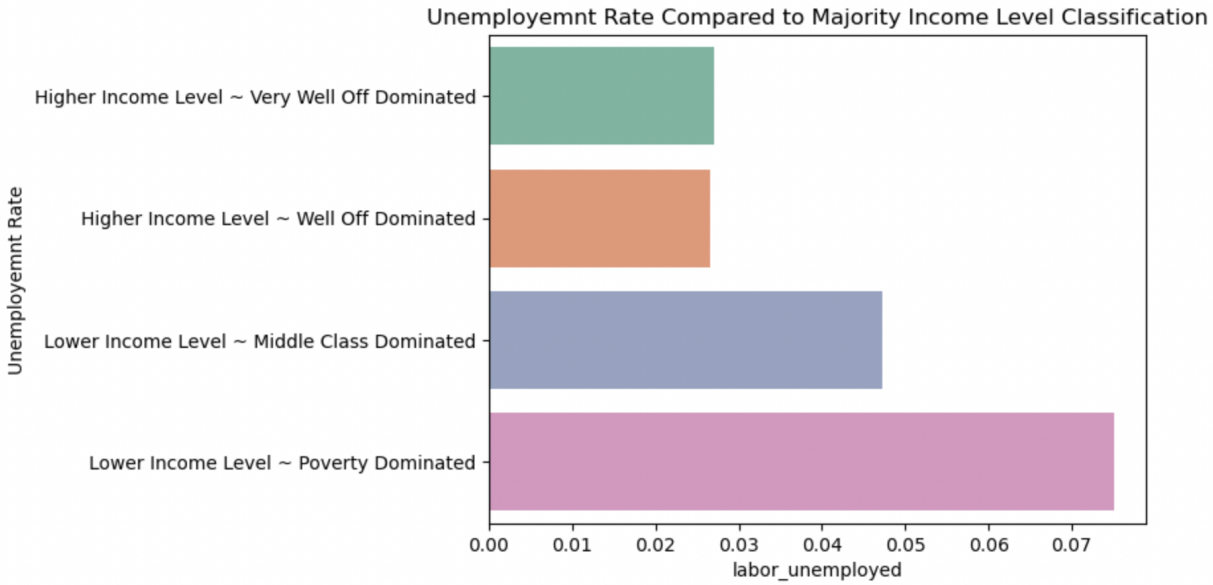
Result #16 - How does the rate of unemployment compare across districts with various percentages of certain income levels?

Skill: Use of groupby and aggregate functions

In this question, we are looking to find what the average unemployment rate is depending on if an area is dominated by ‘poverty’, ‘middle class’, ‘well off’, or ‘very well off”.

Unemployment can vary based on nearly innumerable factors. One thing that may not commonly be considered is what the overall wealth level is in the area with X unemployment rate. We are looking to discover what the correspondence may be between those two factors in this question. If areas with a higher average income have lower unemployment, that could indicate that this is an institutional problem where the rich keep getting richer (due to more access to jobs).

This problem took us a decent amount of code to complete, mainly consisting of functions. We used functions to decipher whether an area had higher income (over 75k) or lower (under 75k), from there, we classified if it was in the higher/lower end of that bucket. Then from there, we grouped by our new category and aggregated the unemployment mean.



We found that unemployment consistently is lower in wealthier neighborhoods. It went almost in perfect order from lowest income highest unemployment to highest income lowest unemployment except for the “well off” category and “very well off” category being neck and neck. This result leads us to believe that individuals in areas with higher incomes have more opportunities to get hired. This discrepancy is more than likely caused from institutional flaws that limit the resources of those less well off.

Off of the last point, we would like to look into the data for the children of individuals in higher vs lower income areas and unemployment rates. It is important to show how high income upbringing can lead to better outcomes (employment in this case).

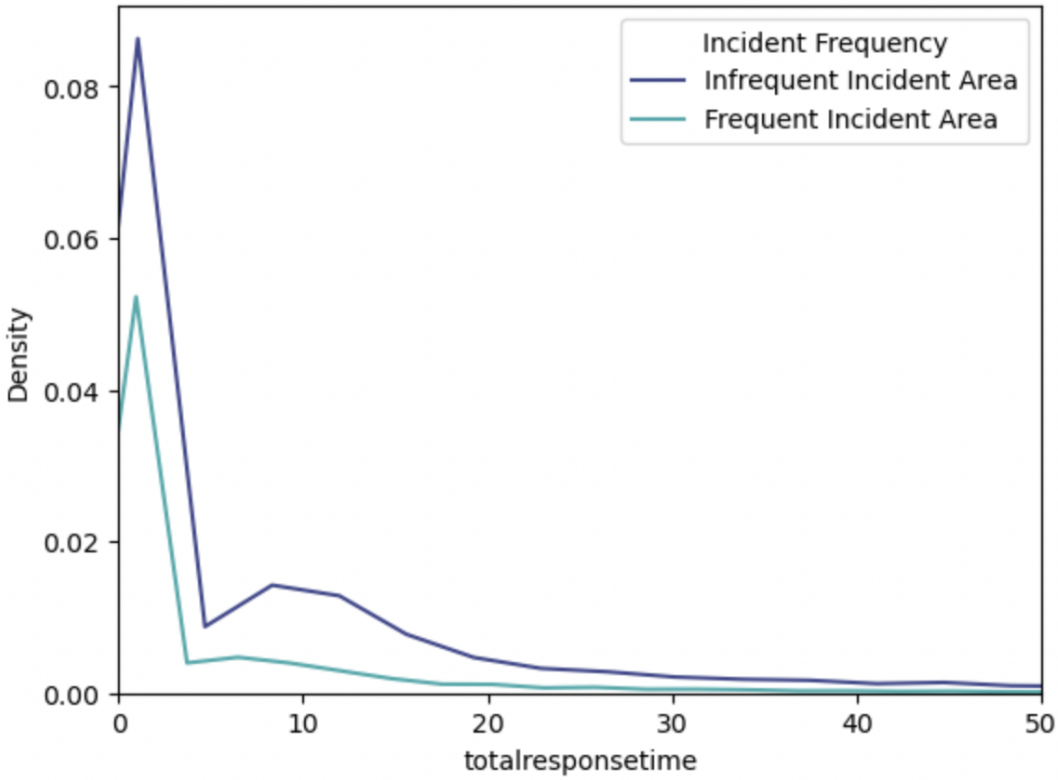
Result #17 - How does total response time vary in neighborhoods with low vs high incidents?

This question was built to look at the differences in total response time in regards to how frequently an area has an incident called into 911.

It is important to answer this question to understand what differences in response times different neighborhoods may have. If a neighborhood has less incidents reported, this could lead to slower response time when they do have incidents. Or, on the other hand, it could lead to faster response time because that neighborhood may take priority. Overall, we were unsure which way it would lean and wanted to uncover potential injustices in response time throughout different neighborhoods.

There were a lot of neighborhoods in the data set so we filtered the top 5 neighborhoods in terms of incidents reported. Oppositely, we filtered 100 neighborhoods from the lower end (less than 1 incident a day). (We used 100 vs. 5 because we needed similar row sample sizes and

the last 100 rows were less than 1 incident a day). Finally, we compared the infrequent vs frequent total response times.



The kernel density plot shows that in neighborhoods, frequent incidents or not, total response time is distributed very similarly and is quite fast at that. There is a higher concentration of slower response times in the infrequent areas but most are quick. The second peak in the dark blue graph shows that some incidents take longer in the infrequent neighborhoods but there may be a reason for this. It can be inferred those crimes may be less serious and require less immediate action. THough, this can't be proven from the data that were made available to us.

The incident type being factored into the total response time would be interesting to look at compared to the regular graph. Serious crimes should have faster response times but maybe that isn't the case, or some neighborhoods have faster response times to certain crimes, this would be hard to quantify but really interesting to visualize.

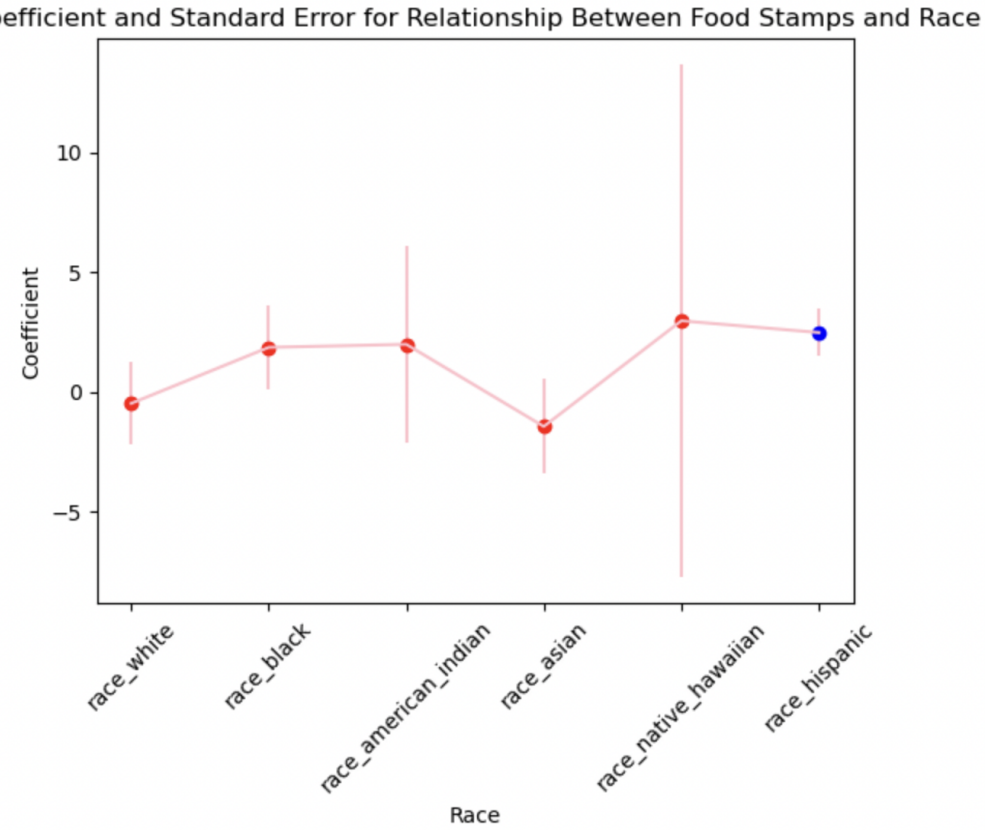
Result #18 - Is there correlation between racial demographics and food stamp usage?

Skill: Logistic regression analysis involving more than 5 independent variables

We want to see if there is any possible (logistic) correlation between the proportions of race and the proportion of the population that uses food stamps.

If there are any discrepancies in which race corresponds with food stamp necessity, this is a serious systemic issue. Food stamps being correlated with one race more than another means that that respective race may be disadvantaged in ways that need to be addressed. Though just correlation between food stamps and race may seem unimportant, it most likely is a clue to causation issues (such as unequal opportunities, unemployment, etc.).

Our code for this problem consisted of logistic regression. The dependent variable was snap (food stamp proportion) and the proportion of different races were the independent variables. We took the logistic regression summary and transformed the data to a data frame which could be plotted. We looked at the coefficients and their standard error to show what the relative slope was. Additionally we showed which values were found to be statistically significant with blue dots.



We found that the races white and asian had negative coefficients (meaning the higher the population proportion the less food stamp proportion) and the races black, american indian, native hawaiian, and hispanic had positive coefficients. This result did not surprise us but is still important. We did find, however, that only the data point for Hispanic people is statistically significant ($p>.05$). We were surprised that there weren't more races that were significant as we have found race to have lasting effects from past historical biases in all of our analysis. But as for the population we did find the correlation to be statistically significant for, we found data to support this finding. More than 5 million Latinos receive food stamps each month which

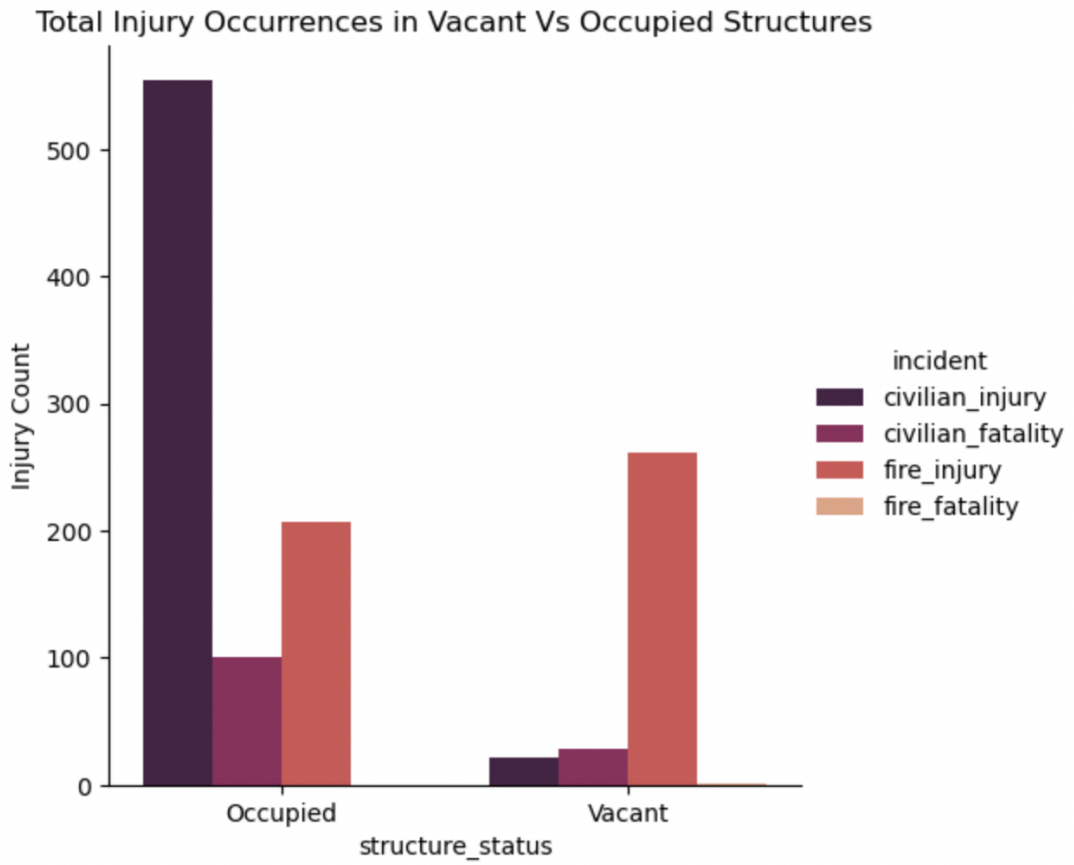
represents more than 10 percent of the Latino population (Center on Budget and Policy Priorities). This data emphasizes that there is a racial bias to food stamps which quite frankly shouldn't exist. Our analysis as a whole has uncovered a lot of injustices in SE michigan.

Result #19 - How do different injury occurrences vary based on the structure occupation status?

Here, we are looking at the different (total count) levels of injuries that occurred during a fire based on the structure state of the building.

We asked this question to get a deeper understanding of how a building's state (vacant or occupied) can impact an individual's harm received from the fire. If we can understand how a building's occupation can affect civilians and firefighters in harm received, we can understand how to combat the fire. If there are significant varying levels of harm, maybe the way a fire is approached should be different in different building types. But, maybe this isn't the case and it would be a waste of time and funding. Our question aims to solve this conundrum.

The code for this problem consisted of grouping a building by its state and getting the sum of all incidents in each injury category (civilian injury, civilian fatality, fire injury, fire fatality). A new data frame was created from that groupby object and we melted it to create 8 different points that would be graphed with continuity. We then used a cat plot to show side by side comparisons of the total incident types in each structure status.



The plot shows us that in occupied structures, civilian deaths and injuries are notably greater than in vacant structures. On the other hand, firefighter injuries are more common in vacant structures than occupied structures. Additionally, compared to the incidents in the vacant category, the total firefighter injury was at least five fold that of the other incidents. This answers our question with an obvious answer, fires in structures with people in them are more harmful to civilians. But surprisingly, firefighters are generally having more incidents in vacant buildings which is hard to explain.

We would want to look at building layouts to further this question. If buildings are less stories, less walls, etc. how does that affect the incidents that occur in different building types?

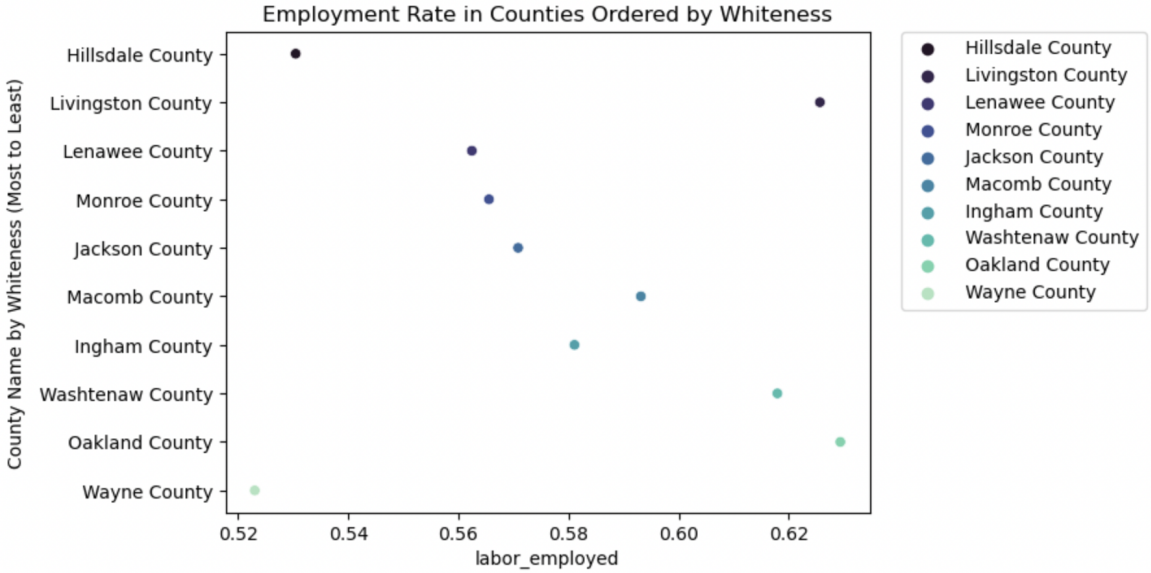
Result #20 - How does whiteness affect employment rates in counties?

In this question, we are comparing employment rates in different neighborhoods. Then, we are looking at how whiteness (on average) affects the employment rate in that neighborhood.

There are a lot of questions we covered about both un/employment and/or race. They all have a general purpose of better understanding inequalities that may exist in our own backyard. This question in particular is important to answer because it could show injustices in hiring

processes or show that progress has been made in the past 60 years. Additionally, understanding how employment can overlap with factors, such as race, can reflect a greater state in a city.

This problem required using a function to scrape the county name from the tract. From that county name, we grouped by that factor and found the average white population across that county. Similarly, we grouped by county name and found the average employment rate within the county and created a dataframe of the variables we wanted to graph. We used an ordered version of the whiteness groupby object to order our data frame and finally plotted average employment rates in counties.



The scatter plots show that generally the employment rate increases as the whiteness proportion decreases. There are a couple exceptions, Livingston County (second most white second most employed) and Wayne County (least white least employed) which would be interesting to look into. With this being shown, in SE Michigan, the more white a county is on average, the lower the employment rate is on average. This could be due to a lot of factors, including individuals choosing not to work (i.e. stay at home mom, white teens not having jobs while their counterparts may, etc.), motivation, necessity, and a lot of other factors.

We would like to look at the same y axis counties compared to the unemployment rate to see if that is consistent with the employment rate trends. Those graphs compared may show contradicting results that illustrate inequalities between races.

References

Facts about Latinos in the food stamp program. Center on Budget and Policy Priorities. (2007, April 19). Retrieved April 20, 2023, from https://www.cbpp.org/research/facts-about-latinos-in-the-food-stamp-program

"Fire Response Time." FIRE RESPONSE TIME, https://fems.dc.gov/page/fire-response-time#:~:text=NFPA%20Standard%201710%20establishes%20a,90%25%20of%20these%20type%20incidents.

Harding, Haley, and George Hunter. "Detroit Remains among Nation's Most Violent Big Cities, FBI Statistics Show." *The Detroit News*, The Detroit News, 28 Sept. 2021, https://www.detroitnews.com/story/news/local/michigan/2021/09/27/detroit-most-violent-big-us-cities-fbi-uniform-crime-report-2020/5883984001/.

Hicks, Mark. "Crews Battle 130-Acre Wildfire at Camp Grayling in Northern Michigan." The Detroit News, The Detroit News, 10 June 2021, https://www.detroitnews.com/story/news/local/michigan/2021/06/09/crews-battle-130-acre-wildfire-camp-grayling-northern-michigan/7630967002/.

Karageorge, E. (n.d.). *Covid-19 recession is tougher on women : Monthly labor review*. U.S. Bureau of Labor Statistics. Retrieved April 20, 2023, from https://www.bls.gov/opub/mlr/2020/beyond-bls/covid-19-recession-is-tougher-on-women.htm#:~:text=In%20past%20recessions%2C%20men%20have,trade%2C%20transportation%2C%20and%20utilities.

"Manchester Police Public Data Dashboard." *City of Manchester NH Official Website*, https://www.manchesternh.gov/Departments/Police/Administrative-Division/Crime-Data/Police-Public-Data-Dashboard.

Appendix

Sydney Lauer's Code

April 18, 2023

In [1]:
```
import seaborn as sns
import numpy as np
import pandas as pd
from collections import Counter
import statsmodels.formula.api as smf
from matplotlib import pyplot as plt
from wordcloud import WordCloud
import geopandas as gpd
```

```
/usr/local/lib/python3.8/site-packages/geopandas/_compat.py:84: UserWarning: The Shapely GEOS
  warnings.warn(
```

In [2]:
```
demographics_df = pd.read_csv("acs_demographics.csv")
income_df = pd.read_csv("acs_income.csv")
work_df = pd.read_csv("acs_work.csv")
calls911_df = pd.read_csv("Detroit-911_Calls_for_Service.csv")
fires_df = pd.read_csv("Detroit-Fire_Incidents.csv", low_memory=False)
```

0.1 What types of 911 calls occur most frequently?

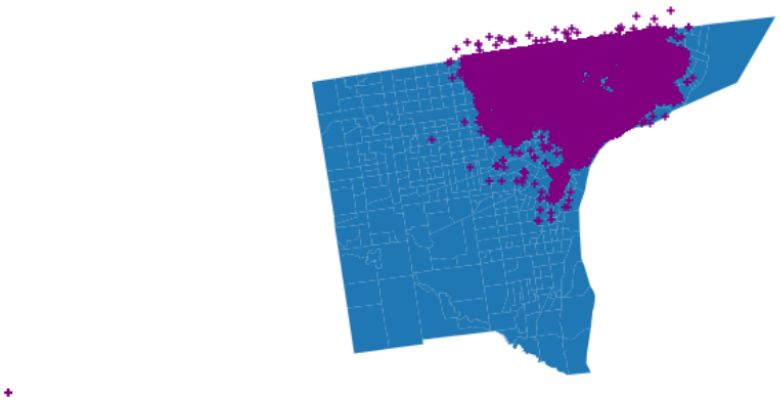
In [3]:
```
counts = Counter()

def count_descriptions(x):
    counts[str(x).lower()] += 1

calls911_df.calldescription.apply(count_descriptions)

wordcloud = WordCloud(width=600,
                      height=400,
                      random_state=2,
                      max_font_size=100)
wordcloud.generate_from_frequencies(counts)
plt.title("World Cloud of 911 Calls")
plt.axis("off")
plt.imshow(wordcloud, interpolation='bilinear')
```

Out[3]:
```
<matplotlib.image.AxesImage at 0x7f7cd0c825b0>
```

0.2 Where within Detroit do calls to 911 occur most frequently?

In [4]:
```
mi_census_tracts = gpd.read_file('tl_2020_26_tract.shp')
detroit_census_tracts = mi_census_tracts[mi_census_tracts["COUNTYFP"] == "163"]

calls_geo = gpd.GeoDataFrame(calls911_df, geometry=gpd.points_from_xy(calls911_df.long
calls_geo.crs = "EPSG:4269"
projection = "+proj=laea +lat_0=30 +lon_0=-95"

fig, ax = plt.subplots(1, figsize=(11,8.5))
ax.axis('off')
detroit_census_tracts.to_crs(projection).plot(ax=ax)
calls_geo.to_crs(projection).plot(ax=ax, marker='+', color='purple')
```

Out[4]:
```
<AxesSubplot:>
```

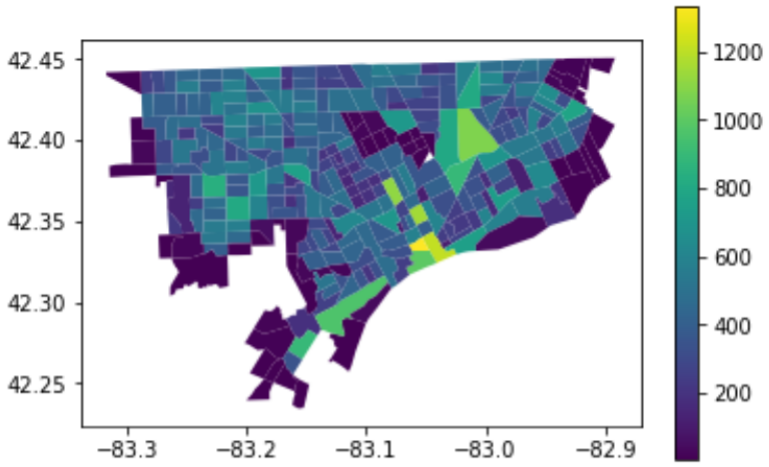


0.3 Which areas of Detroit have more fires on average?

```
In [5]: fires_geo = gpd.GeoDataFrame(fires_df, geometry=gpd.points_from_xy(fires_df.x, fires_d
        fires_geo.crs = "EPSG:4269"

        gdf = gpd.sjoin(detroit_census_tracts, fires_geo, how='inner', op='contains')
        series = gdf.groupby("TRACTCE")["NAME"].count()
        rest_tract_dict = series.to_dict()
        detroit_census_tracts["fire_count"] = detroit_census_tracts["TRACTCE"].map(rest_tract_
        detroit_census_tracts.plot(column = "fire_count", legend= True)
```

```
/usr/local/lib/python3.8/site-packages/geopandas/geodataframe.py:853: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/
  super(GeoDataFrame, self).__setitem__(key, value)
```

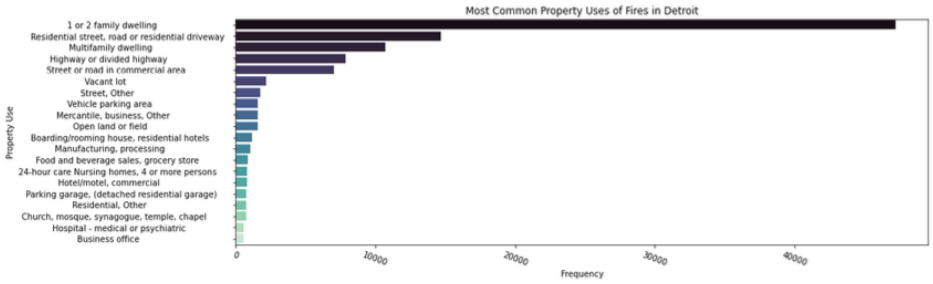
```
Out[5]: <AxesSubplot:>
```



0.4 At what type of locations do fires occur most frequently in detroit?

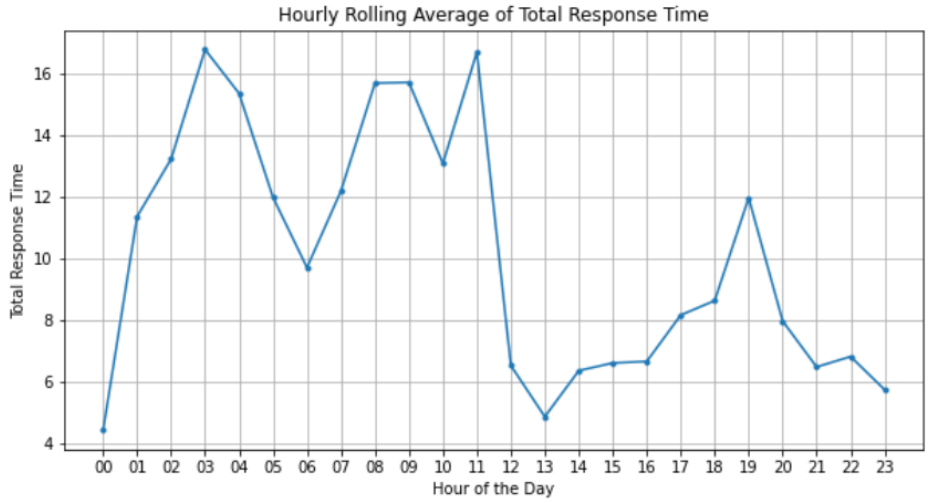
In [6]:
```
c = Counter()

def count_property(x):
    c[str(x)] += 1

fires_df.property_use.apply(count_property)

b = pd.DataFrame(c.most_common(20))
plt.figure(1, figsize=(15,5))
ax = sns.barplot(data = b, x = 1, y = 0, palette="mako")
ax.set_ylabel("Property Use")
ax.set_xlabel("Frequency")
plt.title('Most Common Property Uses of Fires in Detroit')
plt.xticks(rotation = -20)
```

Out[6]:
```
(array([    0., 10000., 20000., 30000., 40000., 50000.]),
 [Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, '')])
```



0.5 Does it take longer for 911 to arrive at certain times of the day?

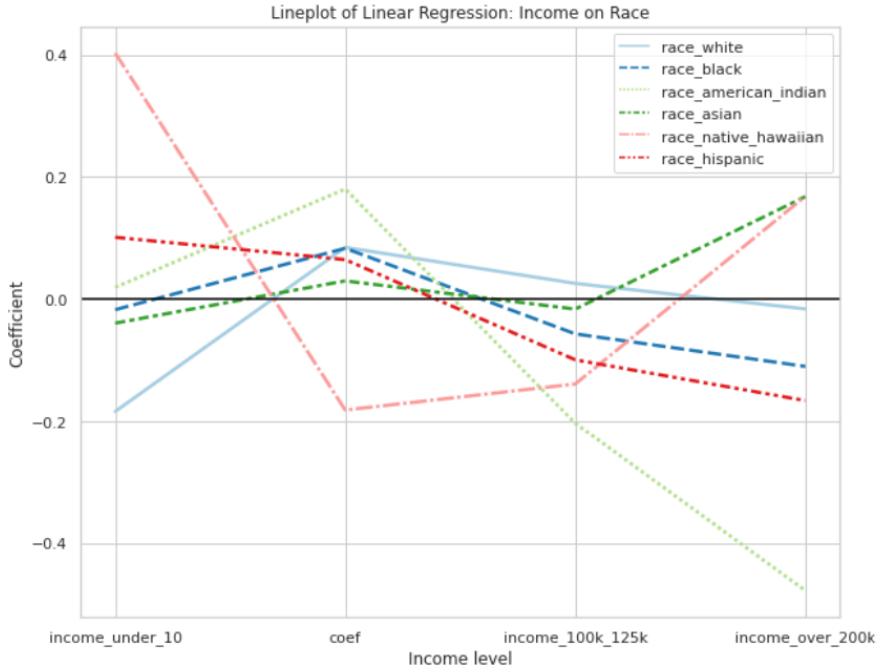
In [7]:
```
def get_hour(x):
    return str(x)[11:13]

calls911_df["call_hour"] = calls911_df.call_timestamp.apply(get_hour)
no_na = calls911_df.dropna()
d = no_na.set_index("call_timestamp").groupby("call_hour").rolling(5).totalresponset
    .groupby("call_hour").mean()
plt.figure(1, figsize=(10,5))
plt.plot(d, marker = '.')
plt.xlabel('Hour of the Day')
plt.ylabel('Total Response Time')
plt.title('Hourly Rolling Average of Total Response Time')
plt.grid(True)
```



0.6 How do racial demographics affect income?

In [8]:
```
race_income_df = income_df.merge(demographics_df, how = "outer", on = "tract")

result_under_10 = ((smf.ols("""income_under_10k ~ race_white + race_black + race_ameri
                + race_native_hawaiian + race_hispanic""", data = race_income_df)).fit
html1 = result_under_10.tables[1].as_html()
under10 = pd.read_html(html1, header=0, index_col=0)[0].reset_index()[["index", "coef"]
under10 = under10.rename(columns = {"coef": 'income_under_10'})

result_50_60 = ((smf.ols("""income_under_10k ~ race_white + race_black + race_american_
                + race_native_hawaiian + race_hispanic""", data = race_income_df)).fit
html2 = result_50_60.tables[1].as_html()
fiftySixty = pd.read_html(html2, header=0, index_col=0)[0].reset_index()[["index", "co

result_100_120 = ((smf.ols("""income_under_10k ~ race_white + race_black + race_americ
                + race_native_hawaiian + race_hispanic""", data = race_income_df)).fit
html3 = result_100_120.tables[1].as_html()
hundredTwenty = pd.read_html(html3, header=0, index_col=0)[0].reset_index()[["index",

result_over_200 = ((smf.ols("""income_under_10k ~ race_white + race_black + race_ameri
                + race_native_hawaiian + race_hispanic""", data = race_income_df)).fit
overTwoHun = pd.read_html(html4, header=0, index_col=0)[0].reset_index()[["index", "co

df1 = under10.merge(fiftySixty, on = "index")
df1 = df1.rename(columns = {"coef_x": 'income_under_10', "coef_y":"income_50k_60k"})
df2 = hundredTwenty.merge(overTwoHun, on = "index")
df2 = df2.rename(columns = {"coef_x": 'income_100k_125k', "coef_y":"income_over_200k"}
df = df1.merge(df2, on = "index")
df = df.transpose().rename(columns = {1: "race_white", 2: "race_black", 3: "race_ameri
                                      4: "race_asian",  5: "race_native_hawaiian", 6:

df = df.drop("index", axis = 0)
df = df.drop(0, axis = 1)

fig = plt.figure(1, figsize=(10,8))
sns.set(font_scale=1)
sns.set_theme(style = "whitegrid")
ax = sns.lineplot(data = df, palette="Paired", linewidth=2.5)
plt.axhline(y=0, color = "black")
ax.set(title='Lineplot of Linear Regression: Income on Race')
ax.set(ylabel = "Coefficient")
ax.set(xlabel = "Income level")
```

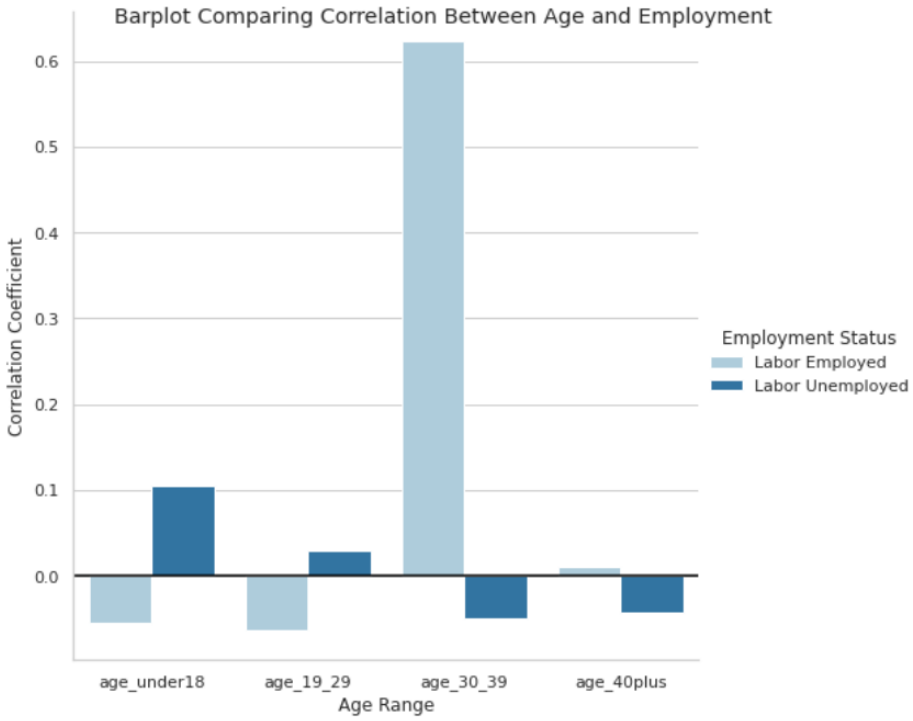
Out[8]:
```
[Text(0.5, 0, 'Income level')]
```



1 Is there a correlation between tracts with older populations and employment levels?

In [9]:
```
age_work_df = work_df.merge(demographics_df, how = "outer", on = "tract")

model_1 = smf.ols('labor_employed ~ age_under18 + age_19_29 + age_30_39 + age_40plus',
result_1 = model_1.fit()
summary_1 = result_1.summary()
html_1 = summary_1.tables[1].as_html()
df_1 = pd.read_html(html_1, header=0, index_col=0)[0].reset_index()
df_1["employment_status"] = "Labor Employed"

model_2 = smf.ols('labor_unemployed ~ age_under18 + age_19_29 + age_30_39 + age_40plus
result_2 = model_2.fit()
summary_2 = result_2.summary()
html_2 = summary_2.tables[1].as_html()
df_2 = pd.read_html(html_2, header=0, index_col=0)[0].reset_index()
df_2["employment_status"] = "Labor Unemployed"
```

```
df = pd.concat([df_1, df_2])
df = df[df["index"] != "Intercept"]

b = sns.catplot(
    data=df, kind="bar",
    x="index", y="coef", hue="employment_status", palette="Paired", height = 7)
plt.axhline(y = 0, color = "black")
b.set_axis_labels("Age Range", "Correlation Coefficient")
b.fig.suptitle('Barplot Comparing Correlation Between Age and Employment')
b.legend.set_title("Employment Status")
```



Lauren's code

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

#demographics_df = pd.read_csv("acs_demographics.csv")
#income_df = pd.read_csv("acs_income.csv")
#work_df = pd.read_csv("acs_work.csv")
calls911_df = pd.read_csv("Detroit-911_Calls_for_Service.csv")
fires_df = pd.read_csv("Detroit-Fire_Incidents.csv")
```

```
<ipython-input-6-6a74280f10f9>:9: DtypeWarning: Columns (14) have mixed types. Specify dtype option on import or set low_memo
  fires_df = pd.read_csv("Detroit-Fire_Incidents.csv")
```

QUESTION 8 average time between call and dispatch for each engine area

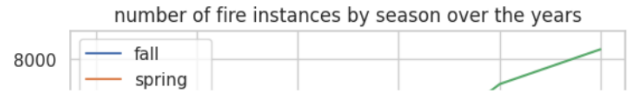
```
import datetime

fires_df['call_timestamp'] = fires_df['call_datetime'].apply(pd.to_datetime)
fires_df['dispatch_timestamp'] = fires_df['dispatch_datetime'].apply(pd.to_datetime)

fires_df['sec_diff'] = (fires_df['dispatch_timestamp'] - fires_df['call_timestamp']) / pd.Timedelta(seconds=1)

engine_area_avg_dispatch = fires_df.groupby('engine_area').sec_diff.mean().reset_index().sort_values('sec_diff')

def nfpa(num):
  if num > 320.00:
    return 'not in standard'
  else:
    return 'in standard'


engine_area_avg_dispatch['NFPA_standard'] = engine_area_avg_dispatch['sec_diff'].apply(nfpa)

fig = plt.figure(1, figsize=(10,12))
sns.barplot(data=engine_area_avg_dispatch, x='sec_diff', y='engine_area', hue='NFPA_standard')
```

```
<Axes: xlabel='sec_diff', ylabel='engine_area'>
```

QUESTION 9 are fires more likely to occur based on what season it is?

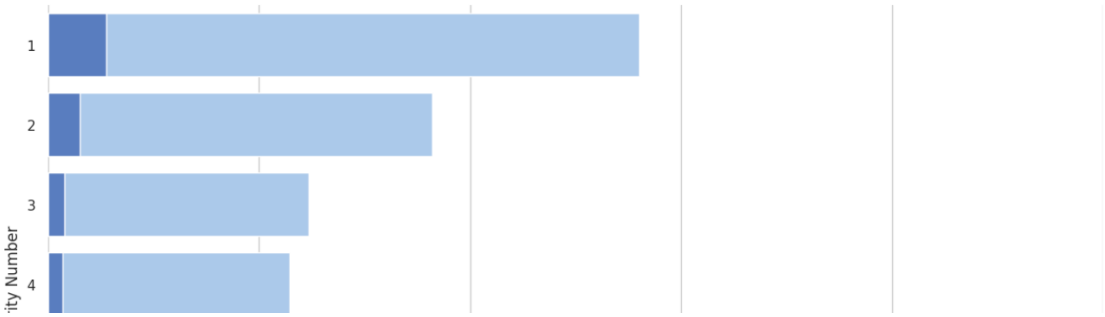
```
# Define a dictionary to map month to season
seasons = {1: 'winter', 2: 'winter', 3: 'spring', 4: 'spring', 5: 'spring',
           6: 'summer', 7: 'summer', 8: 'summer', 9: 'fall', 10: 'fall', 11: 'fall', 12: 'winter'}


def map_month_to_season(row):
    data = row['call_datetime']
    month_tokens = data.split('/')
    month = month_tokens[1]
    month = int(month)
    season = seasons[month]
    return season


fires_df['season'] = fires_df.apply(lambda row: map_month_to_season(row), axis=1)
#display(fires_df.head())

def map_date_to_year(row):
    data = row['call_datetime']
    year_tokens = data.split('/')
    year = year_tokens[0]
    year = int(year)
    return year


fires_df['year'] = fires_df.apply(lambda row: map_date_to_year(row), axis=1)
#display(fires_df.head())

season_counts = fires_df.groupby(['year', 'season']).size().unstack()

for season in season_counts.columns:
    plt.plot(season_counts.index, season_counts[season], label=season)

plt.legend()
plt.xlabel('year')
plt.ylabel('number of fires')
plt.title('number of fire instances by season over the years')
```

```
Text(0.5, 1.0, 'number of fire instances by season over the years')
```



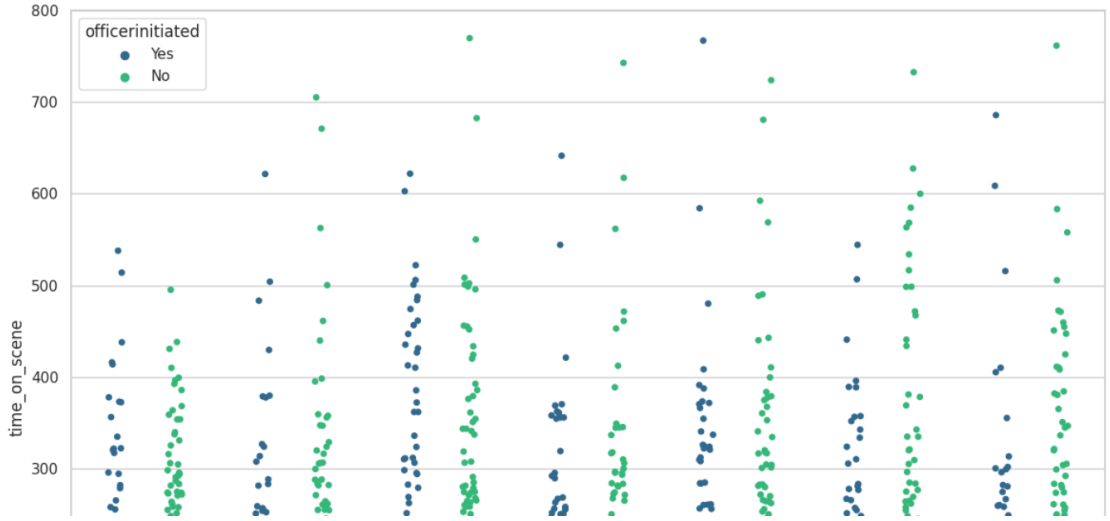
QUESTION 10 How much of the total time does travel take up depending on call priority?

```
priority_response1 = calls911_df.groupby('priority').traveltime.mean().reset_index()

priority_response2 = calls911_df.groupby('priority').totaltime.mean().reset_index()

priority_df = priority_response1.merge(priority_response2, on='priority')
priority_df = priority_df.tail(-1)

sns.set_theme(style="whitegrid")

# Initialize the matplotlib figure
f, ax = plt.subplots(figsize=(15, 8))

sns.set_color_codes("pastel")
sns.barplot(x="totaltime", y="priority", data=priority_df,
            label="Total Time", color="b")


sns.set_color_codes("muted")
sns.barplot(x="traveltime", y="priority", data=priority_df,
            label="Travel Time", color="b")

# Add a legend and informative axis label
ax.legend(ncol=2, loc="lower right", frameon=True)
ax.set(xlim=(0, 100), ylabel="Priority Number",
       xlabel="Time")
sns.despine(left=True, bottom=True)
```



```
smaller_df = calls911_df[['time_on_scene', 'officerinitiated', 'council_district']].dropna()

plt.figure(figsize=(14,10))
p = sns.stripplot(x='council_district', y='time_on_scene', data=smaller_df, jitter=True, hue='officerinitiated', dodge=True, palet
p.set(ylim=(0, 800))
```

```
[(0.0, 800.0)]
```



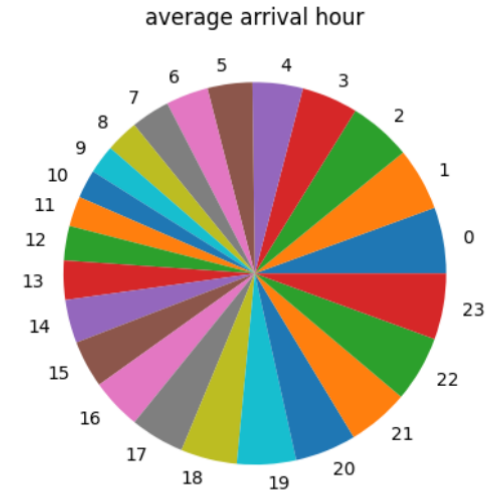
QUESTION 12 does the total time of a 911 emergency change on its zip code?

```
type(calls911_df.zip_code.iloc[0])
calls911_df['zip_as_string'] = calls911_df['zip_code'].apply(lambda x : str(x))
df_12 = calls911_df[calls911_df['totaltime'] >= 0]

list_zc = df_12["zip_code"].unique()[0:15]
new_df = df_12[df_12["zip_code"].isin(list_zc)]

plt.figure(figsize=(25,15))
sns.violinplot(data=new_df, x='zip_as_string', y='totaltime')
plt.xticks(rotation=-90)
plt.title('Total Time on 911 Call per Zip Code');
```

QUESTION 13 What are the common arrival hours for fire fighters.

```
fires_df['arrival_hour'] = fires_df['arrival_datetime'].apply(pd.to_datetime).dt.hour

fires_df = fires_df[fires_df["arrival_hour"].isna() == False]

count_hour = fires_df.groupby('arrival_hour').exposure.count().reset_index()

plt.pie(count_hour['exposure'], labels=count_hour['arrival_hour']);
plt.title('average arrival hour')
```

```
Text(0.5, 1.0, 'average arrival hour')
```



Maddie's code

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

demographics_df = pd.read_csv("acs_demographics.csv")
income_df = pd.read_csv("acs_income.csv")
work_df = pd.read_csv("acs_work.csv")
calls911_df = pd.read_csv("Detroit-911_Calls_for_Service.csv")
fires_df = pd.read_csv("Detroit-Fire_Incidents.csv")

```

```
/var/folders/pq/v5cz250j7rbd9xgxs22d6pp00000gn/T/ipykernel_2171/1469045296.py:8: DtypeWarning: Columns (14) have mixe
d types. Specify dtype option on import or set low_memory=False.
click to expand output; double click to hide output
r("Detroit-Fire_Incidents.csv")
```

In [2]:
```
import plotly.express as px
```

In [213]:
```
import statsmodels.formula.api as smf
import numpy as np
from matplotlib import pyplot as plt
```

How does the level of education affect unemployment rates?

This question wants to find what impact overall education levels have on unemployment. This is useful to answer because it seems like a straight forward question but it is much more complex than people may realize. There is a negative connotation that uneducated people are lazy and some may believe this would lead to higher unemployment. However, that may not statistically be the case (even if it is it doesn't mean people are lazy). By answering this question we can fight to either disprove that idea or give other reasoning if it is the case.

This will help us rach the goal of: "Improve skills in working with unfamiliar semi-structured and structured data". This data is broken up in a way we haven't had the opportunity to fully analyze yet and this question will allow us to break down many data points to use in nour analysis.

For this question, we will have to merge two data sets and run multiple rounds of analysis at each education level. From those rounds of analysis, we can find what the level of education rate at X point has to be to classify the group in that category.

In [5]:
```
def avgEducation(row, level):
    edu = row[level]
    unemployed = row['labor_unemployed']
    if unemployed == 0:
        return 0
```

In [5]:
```
def avgEducation(row, level):
    edu = row[level]
    unemployed = row['labor_unemployed']
    if unemployed == 0:
        return 0
    return edu/unemployed

```

In [6]:
```
# work_df['No education'] = work_df.apply(lambda x: educationOverUnemployed(x, 'edu_none'), axis = 1)
# work_df['Associates'] = work_df.apply(lambda x: educationOverUnemployed(x, 'edu_associates'), axis = 1)
# work_df['Bachelors'] = work_df.apply(lambda x: educationOverUnemployed(x, 'edu_bachelors'), axis = 1)
# work_df['No HS'] = work_df.apply(lambda x: educationOverUnemployed(x, 'edu_no_hs'), axis = 1)
# work_df['Graduate'] = work_df.apply(lambda x: educationOverUnemployed(x, 'edu_graduate'), axis = 1)
```

In [7]:
```
def uneducatedPercentage(row):
    un = row['edu_none'] + row['edu_no_hs']
    return un

work_df['uneducated_unemployment'] = work_df.apply(uneducatedPercentage, axis = 1)
```

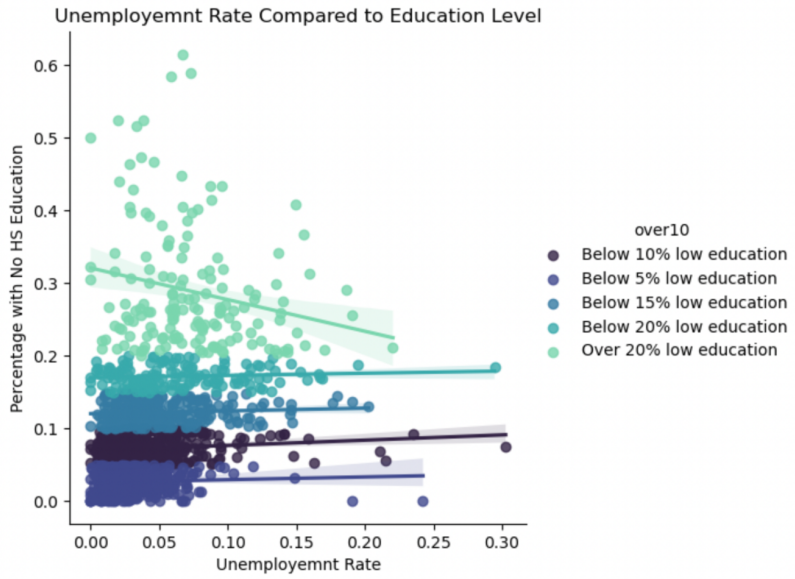
In [8]:
```
def over10Percent(row):
    if (row['edu_none'] + row['edu_no_hs']) < .05:
        return 'Below 5% low education'
    if ((row['edu_none'] + row['edu_no_hs']) < .1):
        return 'Below 10% low education'
    if (row['edu_none'] + row['edu_no_hs']) < .15:
        return 'Below 15% low education'
    if (row['edu_none'] + row['edu_no_hs']) < .20:
        return 'Below 20% low education'
    else:
        return 'Over 20% low education'

work_df['over10'] = work_df.apply(over10Percent, axis = 1)
```

In [9]:
```
# px.scatter(work_df, x=work_df['labor_unemployed'], y=work_df['No education'], color = 'No education')
# px.scatter(x=work_df['labor_unemployed'], y=work_df['Associates'])

```

In [10]:
```
g = sns.lmplot(x="labor_unemployed", y="uneducated_unemployment", data=work_df, hue = 'over10', palette = 'mako')
g.set(ylabel ="Percentage with No HS Education", xlabel = "Unemployemnt Rate",
      title ='Unemplovemnt Rate Compared to Education Level')
```

[10]:
```
g = sns.lmplot(x="labor_unemployed", y="uneducated_unemployment", data=work_df, hue = 'over10', palette = 'mako')
g.set(ylabel ="Percentage with No HS Education", xlabel = "Unemployemnt Rate",
      title ='Unemployemnt Rate Compared to Education Level')
# sns.lmplot(x="labor_unemployed", y="uneducated_unemployment", data=work_df)

```

t[10]: <seaborn.axisgrid.FacetGrid at 0x7f7ae0e8e4d0>



What is the unemployment rate like in more highly female concentrated areas?

In [11]:
```
work_demographics_df = demographics_df.merge(work_df, how='inner', on='tract')
```

In [12]:
```
work_demographics_df.head(1)
```

Out[12]:

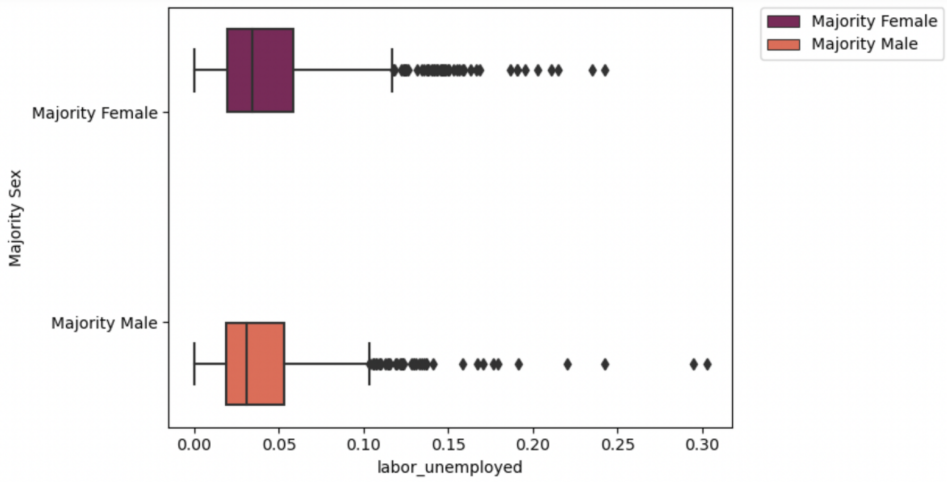
...	labor_unemployed	labor_armed_forces	edu_none	edu_associates	edu_bachelors	edu_no_hs	edu_hs	edu_graduate	uneducated_unemployment	over10
...	0.038335	0.0	0.04494	0.098535	0.129161	0.050266	0.614181	0.062916	0.095206	Below 10% low education

In [13]:
```
def genderDominated(row):
    if row['sex_female'] > .5:
        return 'Majority Female'
    return 'Majority Male'

work_demographics_df['Majority Sex'] = work_demographics_df.apply(genderDominated, axis=1)
```

In [14]:
```
def whichGender(row):
    if row['sex_female'] > row['sex_male']:
        return row['sex_female']
    return row['sex_male']

work_demographics_df['Proportion of Majority Sex'] = work_demographics_df.apply(whichGender, axis=1)
```

In [17]:
```
ax = sns.boxplot(data=work_demographics_df, y="Majority Sex", x="labor_unemployed",
            hue = 'Majority Sex', palette = 'rocket')

ax.legend(bbox_to_anchor=(1.05, 1), loc='upper left', borderaxespad=0)

```

Out[17]: <matplotlib.legend.Legend at 0x7f7ac3f6f1f0>



How does the rate of unemployment compare across districts with various percentages of certain income levels?

In [20]:
```
work_income_df = income_df.merge(work_df, how='inner', on='tract')

# Poverty below 25k
# Middle Class 25k to 75k
# Well Off 75k to 150k
# Very Well Off 150k+
```

In [28]:
```
def poverty(row):
    return row['income_under_10k'] + row['income_10k_15k'] + row['income_15k_20k'] + row['income_20k_25k']

def middleClass(row):
    first_sums = row['income_25k_30k'] + row['income_30k_35k'] + row['income_35k_40k'] + row['income_40k_45k']
    first_sums += row['income_45k_50k'] + row['income_50k_60k'] + row['income_60k_75k']
    return first_sums

def wellOff(row):
    return row['income_75k_100k'] + row['income_100k_125k'] + row['income_125k_150k']

def veryWellOff(row):
    return row['income_150k_200k'] + row['income_over_200k']



work_income_df['Poverty Income Level'] = work_income_df.apply(poverty, axis=1)
work_income_df['Middle Class Income Level'] = work_income_df.apply(middleClass, axis=1)
work_income_df['"Well Off" Income Level'] = work_income_df.apply(wellOff, axis=1)
work_income_df['"Very Well Off" Income Level'] = work_income_df.apply(veryWellOff, axis=1)
```

In [31]:
```
def wellOffAndSome(row):
    lower = row['Poverty Income Level'] + row['Middle Class Income Level']
    if lower > (row['"Well Off" Income Level'] + row['"Very Well Off" Income Level']):
        return "Lower Income Level"
    return "Higher Income Level"

work_income_df['Majority Income Level'] = work_income_df.apply(wellOffAndSome, axis=1)
```

In [34]:
```
def finalBreakDown(row):
    classification = ""
    if row['Majority Income Level'] == "Lower Income Level":
        classification += "Lower Income Level"
        if row['Poverty Income Level'] > row['Middle Class Income Level']:
            classification += " ~ Poverty Dominated"
        else:
            classification += " ~ Middle Class Dominated"
    else:
        classification += "Higher Income Level"
        if row['"Well Off" Income Level'] > row['"Very Well Off" Income Level']:
            classification += " ~ Well Off Dominated"
        else:
            classification += " ~ Very Well Off Dominated"
    return classification

```

```
work_income_df['Income Level Classification'] = work_income_df.apply(finalBreakDown, axis=1)
```

In [44]:
```
graph_work_income = work_income_df.groupby('Income Level Classification')['labor_unemployed'].mean()
```

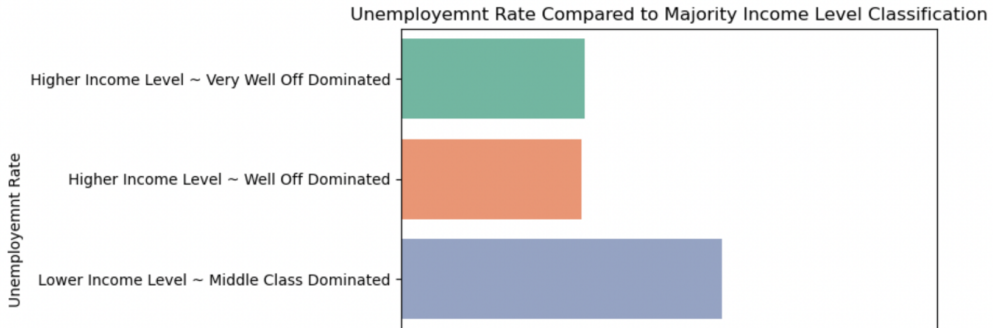
In [50]:
```
graph_work_income_df = pd.DataFrame(graph_work_income).reset_index()
graph_work_income_df
```

Out[50]:

	Income Level Classification	labor_unemployed
0	Higher Income Level ~ Very Well Off Dominated	0.027093
1	Higher Income Level ~ Well Off Dominated	0.026612
2	Lower Income Level ~ Middle Class Dominated	0.047172
3	Lower Income Level ~ Poverty Dominated	0.075117

In [61]:
```
p3 = sns.barplot(y='Income Level Classification', x= 'labor_unemployed', data=graph_work_income_df, palette = 'Set2
p3.set(ylabel = "Unemployemnt Rate",
       title ='Unemployemnt Rate Compared to Majority Income Level Classification')
```

Out[61]:
```
[Text(0, 0.5, 'Unemployemnt Rate'),
 Text(0.5, 1.0, 'Unemployemnt Rate Compared to Majority Income Level Classification')]
```



How does total response time vary in neighborhoods with low vs high incidents?

In [129]:
```
# sns.regplot(x = 'zip_code', y = 'totalresponsetime', line_kws={"color": "black"}, data=calls911_df_clean)
```

In [114]:
```
top_5_frq = list()
for i in range(len(calls911_df_clean.groupby('neighborhood')['council_district'].count().sort_values()[-5:])):
    top_5_frq.append(calls911_df_clean.groupby('neighborhood')['council_district'].count().sort_values()[-5:].index
print(top_5_frq)
```

```
['Wayne State', 'Warrendale', 'Franklin Park', 'Midtown', 'Downtown']
```

In [299]:
```
bottom_100_frq = list()
for i in range(len(calls911_df_clean.groupby('neighborhood')['council_district'].count().sort_values()[:100])):
    bottom_100_frq.append(calls911_df_clean.groupby('neighborhood')['council_district'].count().sort_values()[:100]
# print(bottom_100_frq)
```

In [167]:
```
top_5_bottom_5 = top_5_frq + bottom_25_frq
```

In [168]:
```
 top_5_bottom_5_df = calls911_df_clean[calls911_df_clean['neighborhood'].isin(top_5_bottom_5)]
```

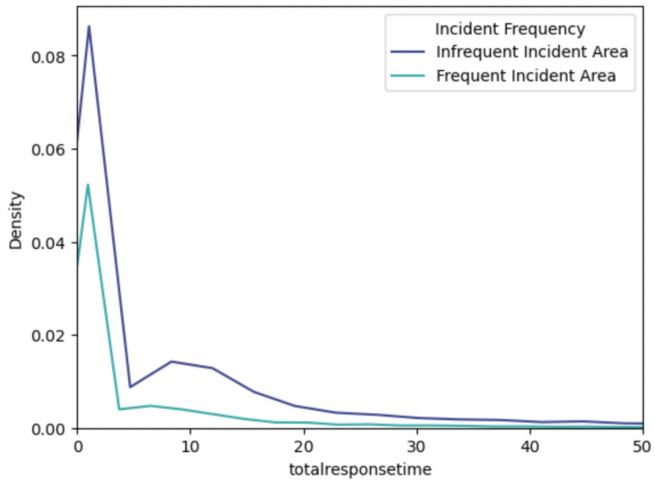
In [169]:
```
def topOrNot(x):
    if x in top_5_frq:
        return "Frequent Incident Area"
    return "Infrequent Incident Area"

top_5_bottom_5_df['Incident Frequency'] = top_5_bottom_5_df['neighborhood'].apply(topOrNot)
```

```
/var/folders/pq/v5cz250j7rbd9xgxs22d6pp00000gn/T/ipykernel_2171/3094458114.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning
-a-view-versus-a-copy
  top_5_bottom_5_df['Incident Frequency'] = top_5_bottom_5_df['neighborhood'].apply(topOrNot)
```

In [197]:
```
# top_5_bottom_5_df[top_5_bottom_5_df['Incident Frequency'] == 'Infrequent Incident Area']
```

In [188]:
```
g = sns.kdeplot(data=top_5_bottom_5_df, x="totalresponsetime", hue="Incident Frequency",
                palette = 'mako', bw_adjust = .2)
g.set(xlim=(0, 50))
```

Out[188]: [(0.0, 50.0)]



Is there correlation between race demographics and food stamp usage?

Type *Markdown* and LaTeX: α^2

In [200]:
```
demographic_income_df = income_df.merge(demographics_df, how='inner', on='tract')
```

In [209]:
```
model = smf.logit('''snap ~ race_white
       + race_black + race_american_indian + race_asian + race_native_hawaiian + race_hispanic'''
                  , data=demographic_income_df).fit()
results_summary = model.summary()
```

```
Optimization terminated successfully.
         Current function value: 0.252880
         Iterations 7
```

In [210]:
```
results_as_html = results_summary.tables[1].as_html()
log_demographic_income_df = pd.read_html(results_as_html, header=0, index_col=0)[0]
pre_reset_index_num = log_demographic_income_df.drop(columns=['[0.025', '0.975]', 'z'])
log_demographic_income_df = pre_reset_index_num.reset_index()
```

In [245]:
```
red_dots1 = log_demographic_income_df[log_demographic_income_df['P>|z|'] > .05][1:]
blue_dots1 = log_demographic_income_df[log_demographic_income_df['P>|z|'] <= .05]
```

In [251]:
```
plt.errorbar(y= 'coef', x = 'index', yerr= 'std err', data = log_demographic_income_df[1:], color = 'pink')
plt.plot('index','coef', data=red_dots1, color='red', marker='o', linestyle='none')
plt.plot('index','coef', data=blue_dots1, color='blue', marker='o', linestyle='none')
plt.xticks(rotation=45)
plt.xlabel('Race')
plt.ylabel('Coefficient')
plt.title('Coefficient and Standard Error for Relationship Between Food Stamps and Race')
```

Out[251]: Text(0.5, 1.0, 'Coefficient and Standard Error for Relationship Between Food Stamps and Race')

How do different injury occurences vary based on the structure occupation status?

In [266]:
```
fires_df_structure_clean = fires_df.dropna(subset = ['structure_status', 'civilian_fatality'])
```

In [298]:
```
# fires_df_structure_clean
```

In [292]:
```
structure_injury_df = fires_df_structure_clean.groupby('structure_status')[['civilian_injury', 'civilian_fatality',
dfs = structure_injury_df.sum().reset_index()
dfs

```

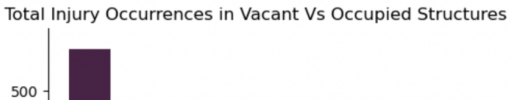
Out[292]:

	structure_status	civilian_injury	civilian_fatality	fire_injury	fire_fatality
0	Occupied	554	101	207	0
1	Vacant	22	28	261	1

In [293]:
```
dfs1 = pd.melt(dfs, id_vars = 'structure_status')
dfs1 = dfs1.rename(columns={"variable": "incident"})
# dfs1
```

In [297]:
```
ax = sns.catplot(x = 'structure_status', y='value',
                 hue = 'incident',data=dfs1, kind='bar', palette = 'rocket')
ax.set(ylabel = "Injury Count",
       title ='Total Injury Occurrences in Vacant Vs Occupied Structures')
```

Out[297]: <seaborn.axisgrid.FacetGrid at 0x7f7ae134d2d0>



How does whiteness affect employment rates in counties?

In [317]:
```
work_demographics_df.columns
```

Out[317]:
```
Index(['state_x', 'county_x', 'tract', 'tract_name', 'pop_unweighted',
       'pop_sex', 'pop_race', 'pop_ethnicity', 'sex_male', 'sex_female',
       'race_white', 'race_black', 'race_american_indian', 'race_asian',
       'race_native_hawaiian', 'race_hispanic', 'age_under18', 'age_19_29',
       'age_30_39', 'age_40plus', 'state_y', 'county_y', 'pop_labor',
       'pop_edu', 'labor_not_in_workforce', 'labor_employed',
       'labor_unemployed', 'labor_armed_forces', 'edu_none', 'edu_associates',
       'edu_bachelors', 'edu_no_hs', 'edu_hs', 'edu_graduate',
       'uneducated_unemployment', 'over10', 'Majority Sex',
       'Proportion of Majority Sex', 'County Name'],
      dtype='object')
```

In [387]:
```
# !pip install pysal
```

In [388]:
```
# %matplotlib inline
# import os
# os.environ['USE_PYGEOS'] = '0'
# import geopandas as gpd
# import pandas as pd
# import pysal
# from collections import defaultdict
# import matplotlib.pyplot as plt
# import matplotlib as mpl
# from matplotlib.pylab import cm

# from esda.getisord import G_Local
# from esda.moran import Moran
# from libpysal.weights import KNN

# from tqdm.auto import tqdm
# import random
# import mapclassify
```

In [314]:
```
def getCountyName(x):
    splitMe = x.split(',')
    return splitMe[1]

work_demographics_df['County Name'] = work_demographics_df['tract_name'].apply(getCountyName)
```

In [373]:
```
county_unemployemnt = pd.DataFrame(work_demographics_df.groupby('County Name')['labor_employed'].mean()).reset_in
```

In [374]:
```
mostWhiteToLeast = (pd.DataFrame(work_demographics_df.groupby('County Name')['race_white'].mean()).reset_index().
```

In [375]:
```
county_unemployemnt['County Name by Whiteness (Most to Least)'] = pd.Categorical(
    county_unemployemnt['County Name'],
    categories= mostWhiteToLeast,
    ordered=True
)
county_unemployemnt.sort_values('County Name by Whiteness (Most to Least)')
```

Out[375]:

	County Name	labor_employed	County Name by Whiteness (Most to Least)
0	Hillsdale County	0.530447	Hillsdale County
4	Livingston County	0.625653	Livingston County
3	Lenawee County	0.562443	Lenawee County
6	Monroe County	0.565554	Monroe County
2	Jackson County	0.570842	Jackson County
5	Macomb County	0.593139	Macomb County
1	Ingham County	0.581081	Ingham County
8	Washtenaw County	0.617927	Washtenaw County
7	Oakland County	0.629360	Oakland County
9	Wayne County	0.523014	Wayne County

In [377]:
```
# pd.DataFrame(work_demographics_df.groupby('County Name')['race_white'].mean()).reset_index()
```

In [390]:
```
ax = sns.scatterplot(data=county_unemployemnt, y="County Name by Whiteness (Most to Least)", x="labor_employed",
            hue = 'County Name by Whiteness (Most to Least)', palette = 'mako')

ax.legend(bbox_to_anchor=(1.05, 1), loc='upper left', borderaxespad=0)
ax.set(title = 'Employment Rate in Counties Ordered by Whiteness')
```