

# EK 125 Project

## Group:

- Parker Dunn
- Brennan Lavoie
- Sam Yard

```
clear all;  
close all;
```

## Section: Introduction

Strokes are the 2nd leading cause of death globally, according to the World Health Organization (WHO). Over 10% of total deaths are the result of a stroke (Fedesoriano). The WHO has called the stroke, the 'incoming epidemic of the 21st century' (Sarikaya, Ferro & Arnold, 2015). A stroke occurs when blood supply to part of the brain is cut off or reduced, preventing oxygen and nutrients from reaching brain cells. Without oxygen or nutrients brain cells begin to die after just minutes. Many strokes are preventable (Guzik & Bushnell, 2017). Some risk factors leaving one more susceptible to a stroke include hypertension (high blood pressure), smoking, diabetes mellitus (which results in elevated levels of blood glucose), a previous stroke, and obesity (Guzik & Bushnell, 2017). If there was a way to predict the likelihood of a patient developing a stroke, how many lives would we be able to save with effective medical interventions and/or lifestyle changes? Our selected dataset outlines this potentially relevant patient information to predict if a patient is more prone to have a stroke.

## References

Fedesoriano (Ed.). (2021, January 26). Stroke prediction dataset. Retrieved April 04, 2021, from <https://www.kaggle.com/fedesoriano/stroke-prediction-dataset>

Guzik, A., & Bushnell, C. (2017). Stroke Epidemiology and Risk Factor Management. Continuum (Minneapolis, Minn.), 23(1, Cerebrovascular Disease), 15–39. <https://doi.org/10.1212/CON.0000000000000416>

Sarikaya, H., Ferro, J., & Arnold, M. (2015). Stroke prevention--medical and lifestyle measures. European neurology, 73(3-4), 150–157. <https://doi.org/10.1159/000367652>

## Section: Loading Data

### Importing the Data and Brief Overview of the Data

Loading in the "*healthcare-dataset-stroke-data.csv*" data.

Make sure to **(1)** have the CSV file in the working directory and **(2)** match the file name used above (or change it below) to ensure this code will load the data.

```
%This version of the "readtable" call only reads in a subset of the file
%{
Commented this out cause we don't really need it

strokedata_sample = readtable("healthcare-dataset-stroke-data.csv", 'Range', 'A1:L5');

%}
```

```
%This version of the "readtable" call loads in the entire stroke dataet
strokedata = readtable("healthcare-dataset-stroke-data.csv");
```

A little overview of the information loaded in "strokedata"

Fields - *Total of 12*

- |                        |   |
|------------------------|---|
| 1. "id"                | - A list of integers to identify the patients |
| 2. "gender"            | - Male or Female                              |
| 3. "age"               | - Integers                                    |
| 4. "hypertension"      | - Binary                                      |
| 5. "heart_disease"     | - Binary                                      |
| 6. "ever_married"      | - Yes/No                                      |
| 7. "work_type"         | - Categorical information - strings           |
| 8. "Residence_type"    | - Categorical (urban or rural)                |
| 9. "avg_glucose_level" | - Float value                                 |
| 10. "bmi"              | - Float, Int, or N/A                          |
| 11. "smoking_status"   | - Categorical                                 |
| 12. "stroke"           | - Binary                                      |

**Setting up a short preview section, including a function that loads any choice of rows and fields.**

**(NOTE: There is no content here that is relevant to our project unless you would like to preview some of the data for some reason. This was useful early on in our project but did not contribute to our general goals.)**

Select the fields you want - enter 1 to extract the field or 0 to ignore the field

```
id_f           = 0;
gender_f       = 0;
age_f         = 1;
hypertension_f = 0;
heart_disease_f = 0;
ever_married_f = 0;
work_type_f    = 0;
```

```

Residence_type_f    = 0;
avg_glucose_level_f = 0;
bmi_f               = 1;
smoking_status_f    = 0;
stroke_f            = 1;

%creating the vector passed to the function preview_data
log_vec = logical([id_f, gender_f, age_f, hypertension_f, heart_disease_f, ever_married_f, work_

```

Select the rows you want.

There are **two approaches you can use to extract rows**.

(1) Select specific rows - see *bullet points below*

- Sections of rows can be passed via colon notation (e.g., 10:1:100)
- The rows will be extracted in numerical order - e.g. if [10, 2, 100, 2000, 60] are requested the rows will be returned in the order [2, 10, 60, 100, 2000].

(2) Select all rows

```

% Enter selected method for extracting rows
% "all rows" = 1
% "specific rows" = 2
choice = 2;

% Enter the specific rows you want below
rowrange = [1:10]; % enter the row numbers in the vector

preview_table = preview_data(strokedata, log_vec, choice, rowrange);
%struct displays below

% PREVIEW DISPLAYS HERE
%disp(preview_table)
% Just clearing up some memory.
% No need to keep these variables.
clearvars id_f gender_f age_f hypertension_f heart_disease_f ever_married_f work_type_f Residence_f
clearvars log_vec rowrange choice preview_table;

```

## Section: Exploratory Data Analysis

First, let's look at what portion of the patients in the study had a stroke or didn't have a stroke.

It is good sampling practice and good for predictive models to consider both people who did and did not have a stroke. Individuals who had a stroke provide insight into common lifestyle choices that may lead to strokes. Meanwhile, the subjects who did not have strokes offer a baseline of comparison. While we may find a common lifestyle choice among subjects who had a stroke, this factor may also be common among people who did not have a stroke (and generally common among the collective participants in the study). This comparison will be valuable in determining predictive factors for strokes.

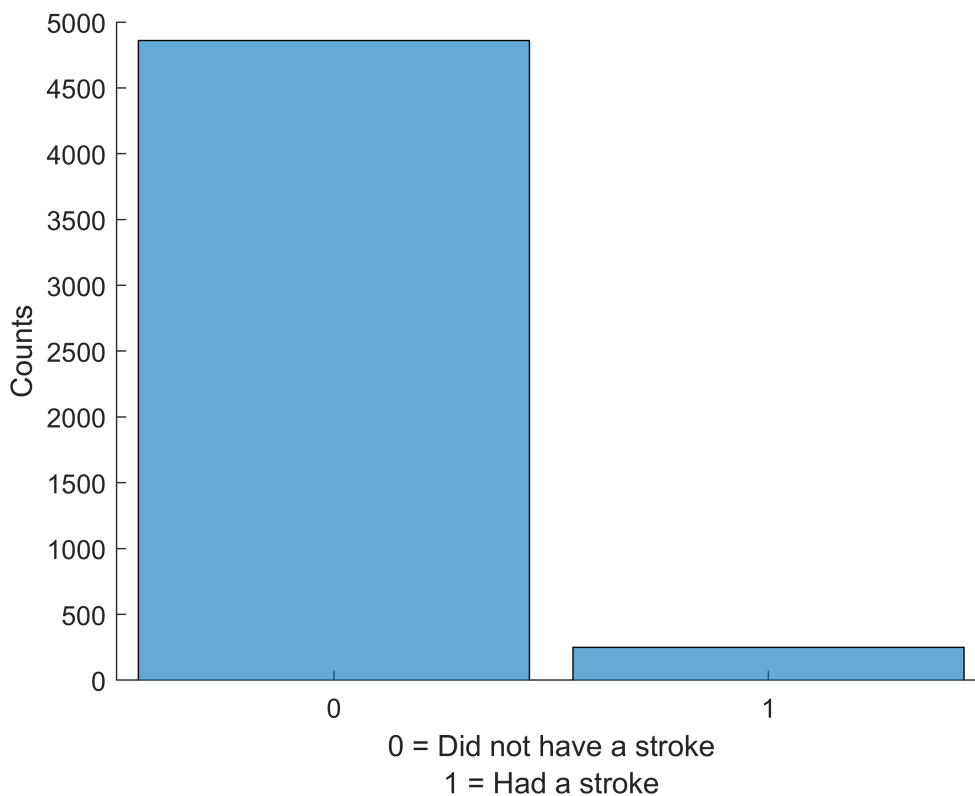
Since it will be beneficial to compare the two groups of stroke patients, the distribution of these patients will be examined first.

```
% Grabbing the "stroke" column of the table
stroke_col_array = strokedata.stroke;

stroke_dist_figure = figure(1);
clf(stroke_dist_figure)
hold on

stroke_dist = histogram(categorical(stroke_col_array),{'0','1'});
xlabel(["0 = Did not have a stroke","1 = Had a stroke"]);
ylabel("Counts")

hold off
```



```
% That count is really low. Determining the exact number of people who had
% a stroke
num_strokes = length(stroke_col_array(stroke_col_array == 1));
fprintf("There are %d individuals in the dataset who had a stroke.",num_strokes)
```

There are 249 individuals in the dataset who had a stroke.

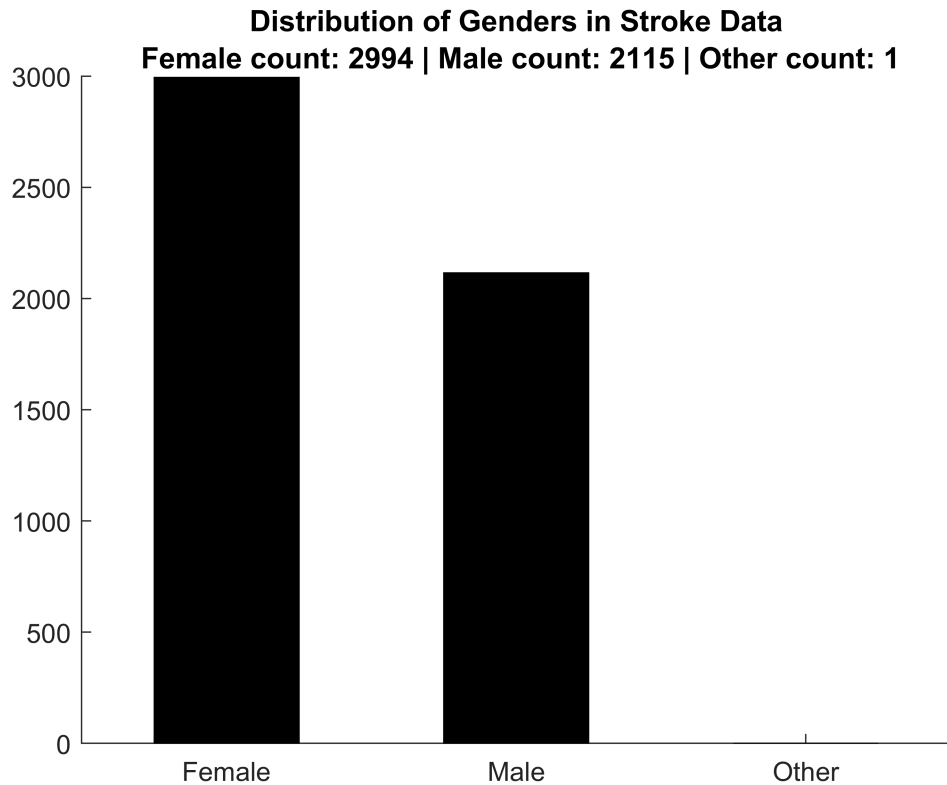
```
clearvars num_strokes stroke_col_array
```

Clearly, more patients were surveyed for this study that did not have a stroke compared to the small group of 249 people that did have a stroke. The sample size of patients with a history of stroke is small compared to the entire collection of people interviewed, which limits, to some degree, the value of any conclusions about strokes that we may develop from this data. The sample is probably a bit small to be able to generate robust predictions about a person's chances of having a stroke. However, the primary focus here is to identify lifestyle features that correlate with risk of stroke. The large sample size of non-stroke patients means that any risk factor or lifestyle choice that clearly distinguishes the group of patients with a stroke from the rest of the patients is highly likely to be predictive of a stroke.

## Second, basic statistics about gender. Only one piece of information is available: how many males and how many females.

```
% Gotta grab the only information about gender.
gender_array = strokedata.gender;

% Determining all of the unique entries in this data
entry_types = unique(gender_array);
% Now that we know what we are looking at, let's get a distribution of
% these entries
gender_categories = categorical({'Female','Male','Other'});
sort_gender_categories = reordercats(gender_categories,{'Female','Male','Other'});
gender_dist_plot = figure(2);
clf(gender_dist_plot) %note: this just clears the figure so changes can be made and the old one
hold on
% grabbing the plotting values
gender_dist = histcounts(categorical(gender_array),sort_gender_categories);
%plotting the information
bar(sort_gender_categories,gender_dist,0.5,'k');
%interesting title for the plot
gender_plot_title = sprintf("Distribution of Genders in Stroke Data\n Female count: %d | Male c
                                gender_dist(1),gender_dist(2),gender_dist(3));
title(gender_plot_title);
hold off
```



```
clearvars gender_plot_title gender_dist gender_categories sort_gender_categories entry_types; %
```

The participants in the study are actually mostly listed as male or female, despite the other category, which simplifies working with this gender data. The general distribution of males and females is nearly 50/50 and nothing substantial stands out about the gender distribution that would suggest there needs to be concerns about working with this data.

### Third, basic statistics about age. An examination of the distribution of ages.

```
%grabbing the age data
age_array = strokedata.age;

%Going to do a subplot
% Subplot 1: All ages ... no binning
% Subplot 2: All ages ... binned within a range
% (ex. bins of 5 years, 21-25 | 26-30 | 31-35 etc.)

age_distribution_plot = figure(3);
clf(age_distribution_plot)

hold on
subplot(1,2,1);
% PLOT 1 Script
no_bins_edges = [0.5:1:99.5]; %Should indicate to "histogram" function what values we want in t
```

```

age_no_bins_hist = histogram(age_array,no_bins_edges,'FaceColor',[0.3, .6, 0.3]);
%Modifying histogram
titleforsubplot1 = sprintf("Distribution of ages without bins\n(each integer age plotted separately)");
title(titleforsubplot1);
xlabel("Ages");

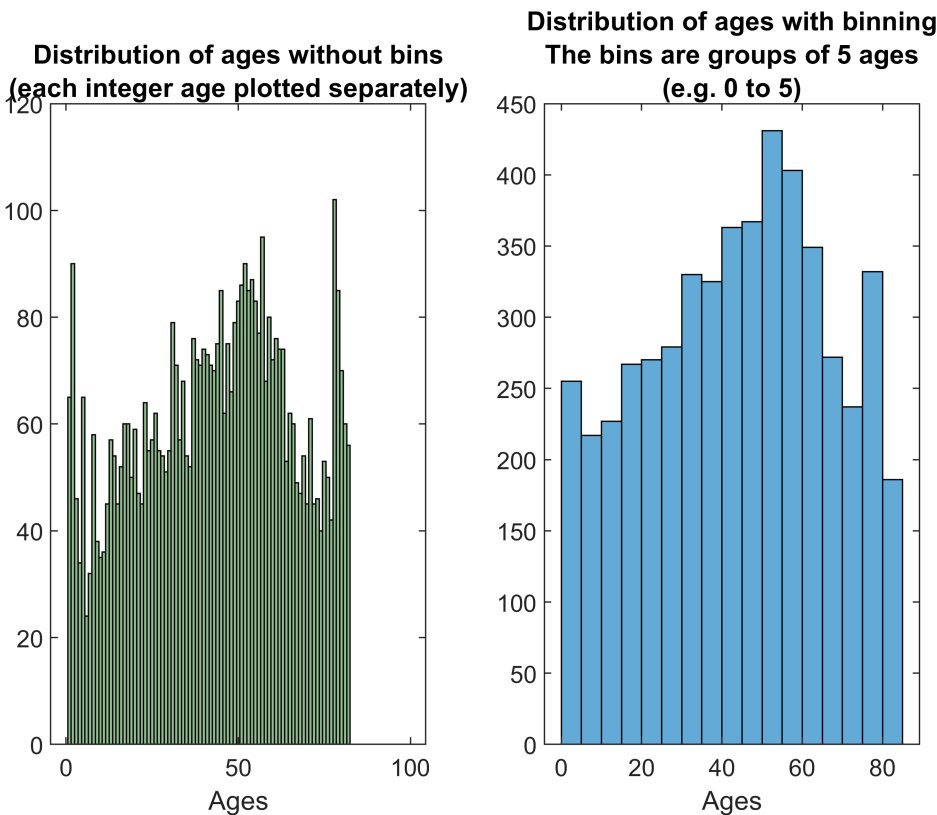
%PLOT 2 Script
subplot(1,2,2);
%going to use features of "histogram" function to do binning
age_bins_hist = histogram(age_array,'BinMethod','auto');

titleforsubplot2 = sprintf("Distribution of ages with binning\nThe bins are groups of %d ages\n",
    (age_bins_hist.BinEdges(2)-age_bins_hist.BinEdges(1)),...
    age_bins_hist.BinEdges(1), age_bins_hist.BinEdges(2));

title(titleforsubplot2)
xlabel("Ages")

hold off

```



```

clearvars age_no_bins_hist titleforsubplot1 titleforsubplot1 no_bins_edges ages_no_stroke ages_
clearvars ages_distribution_plot ages_distribution_stroke_conditional; %freeing up computer space

```

The age distributions reveal some deviations from a normal data distribution. Age does not seem like a feature that would generally follow a normal distribution, but it's still important to keep of the trends in this data among the subjects.

In particular, it is interesting that the distribution of all ages has large spikes at very high and very low ages. The age 1 bin has nearly 100 subjects from the study. The study was conducted among patients in a hospital,

so seeing 100 babies in a hospital is not unexpected since they likely make far more visits to doctors than the average healthy adult, but this group may need to be separated out when it comes to using this data for predicting strokes. **It is unlikely that babies and young children can effectively provide information about predicting the potential for a stroke.**

**Next, we will extract the number of patients who have hypertension and look to see how many of them also had a stroke.**

```
% Exploring the distribution of patients who had a stroke and hypertension

hypertension_mask = (strokedata.hypertension == 1);           % Extract a logical of patients who
count_hypertension = sum(hypertension_mask);                 % total number of people who had a
hypertension_positive = strokedata(hypertension_mask,:);      % table including only patients who
stroke_mask = (hypertension_positive.stroke==1);             % Further extracting parts of the t
count_stroke = sum(stroke_mask);
```

```
percent_hypertension_stroke = (count_stroke/count_hypertension)*100;
```

```
fprintf("There were %d participants who have a history of hypertension and %d of them also suffered a stroke.\n", count_hypertension, count_stroke);
```

There were 498 participants who have a history of hypertension and 66 of them also suffered a stroke.

```
fprintf("This means that %.2f%% of the participants who have hypertension also suffered a stroke.\n", percent_hypertension_stroke);
```

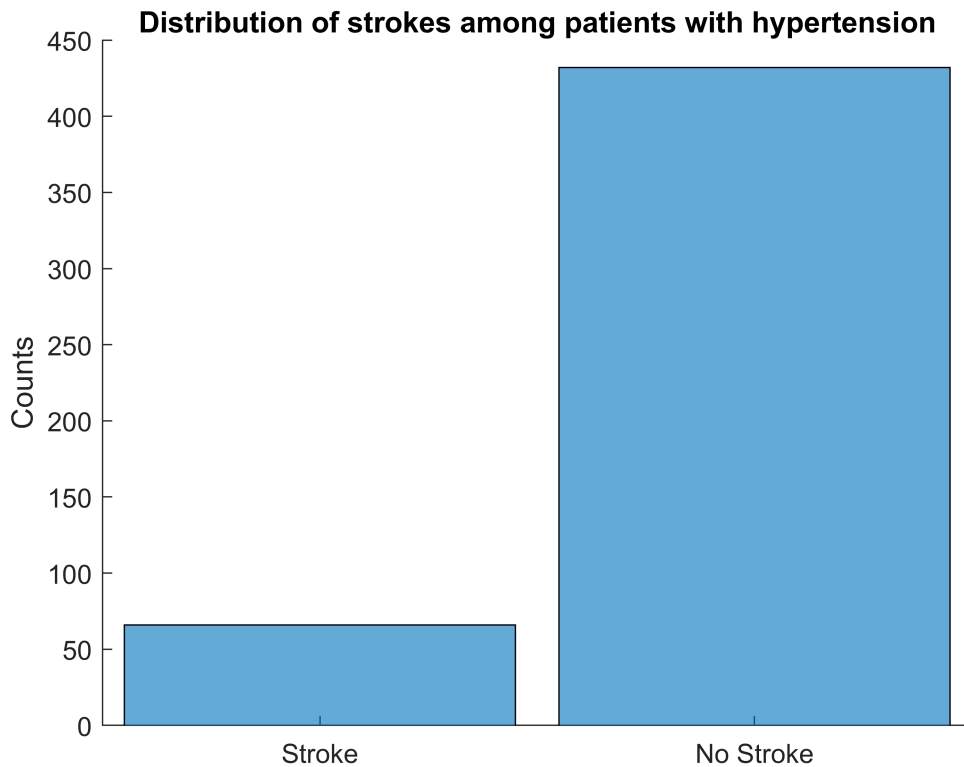
This means that 13.25% of the participants who have hypertension also suffered a stroke.

```
y1 = count_stroke;
y2 = count_hypertension - count_stroke;
figure(21);
clf(21);
hold on

histogram('Categories',{'Stroke','No Stroke'},'BinCounts',[y1, y2])
title("Distribution of strokes among patients with hypertension")
ylabel("Counts")

hold off
```





This data is simply meant to help demonstrate the distribution of patients with hypertension. There are significantly more patients who did not have a stroke than who did, so it is not surprising to see far more non-stroke patients with hypertension. There are two helpful observations that you can make from this data though. First, a relatively low number of patients (only around 10%) reported having hypertension. Second, while the number of stroke patients with hypertension is a low total, more than 20% of the stroke patients had hypertension. It is not definitive information, but there does appear to be a higher rate of hypertension among the stroke patients (more than 20%) compared to the general collection of patients (about 10%).

**Now, we will extract the number of patients who have a history of heart disease and look to see how many of them also had a stroke.**

```
heart_mask = strokedata.heart_disease == 1;
count_heart = sum(heart_mask);
heart_positive = strokedata(heart_mask,:);
heart_mask = heart_positive.stroke==1;
count_heart_stroke = sum(heart_mask);
```

```
percent_heart_disease_stroke = (count_heart_stroke/count_heart)*100;
```

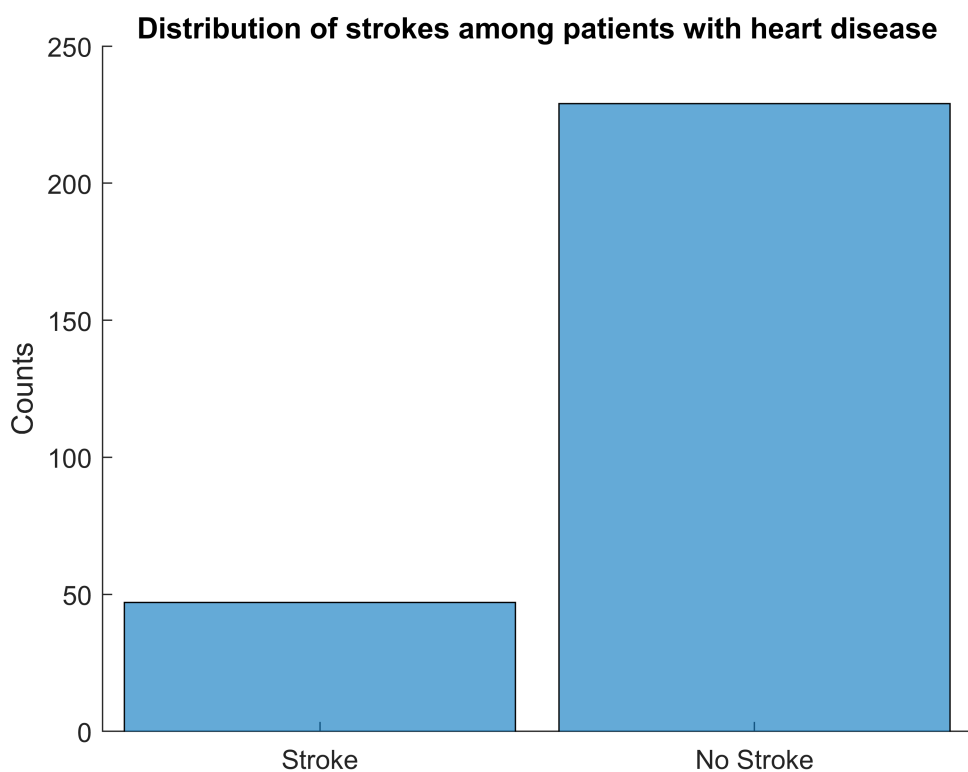
```
fprintf("There were %d participants have a history of heart disease and %d of them also suffered a stroke.\n", count_heart, count_heart_stroke);
```

There were 276 participants have a history of heart disease and 47 of them also suffered a stroke.

```
fprintf("This means that %.2f%% of the participants who have hypertension also suffered a stroke.\n", percent_heart_disease_stroke);
```

This means that 17.03% of the participants who have hypertension also suffered a stroke.

```
yhd1 = count_heart_stroke;  
yhd2 = count_heart - count_heart_stroke;  
  
figure(22);  
clf(22);  
hold on  
  
histogram('Categories',{'Stroke','No Stroke'},'BinCounts',[yhd1, yhd2])  
title("Distribution of strokes among patients with heart disease")  
ylabel("Counts")  
  
hold off
```



```
clearvars count_heart count_heart_stroke heart_mask heart_positive
```

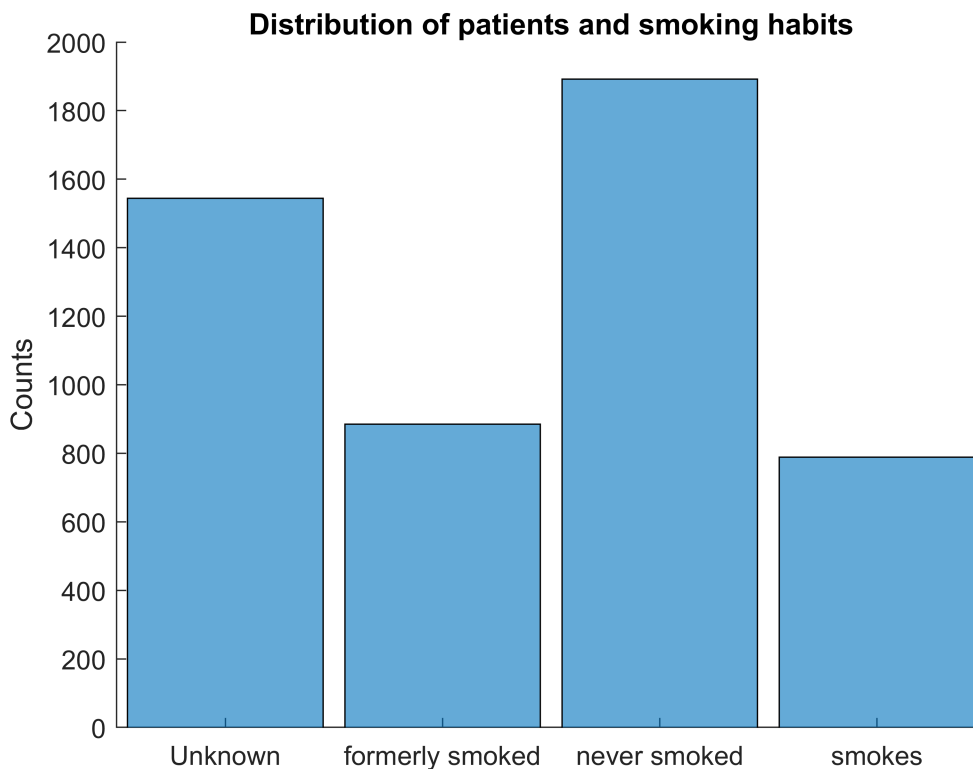
This data is simply meant to help demonstrate the distribution of patients with heart disease. There are significantly more patients who did not have a stroke than who did, so it is not surprising to see far more non-stroke patients with heart disease. There are helpful observations that you can make from this data though. First, a relatively low number of patients (only around 5%) reported having heart disease. Second, while the number of stroke patients with heart disease is a low total, more than about 20% of the stroke patients had heart disease. It is not definitive information, but there does appear to be a higher rate of heart disease among the stroke patients.

## Next, the distribution of strokes among smokers and non-smokers

```
smoking_col_from_table = strokedata.smoking_status;  
smoking_array_entries = unique(smoking_col_from_table); % just returns all the categories that  
[counts, order_of_counts] = histcounts(categorical(smoking_col_from_table),smoking_array_entries);  
for i = 1:length(counts)  
    fprintf("%s: %d counts\n",order_of_counts{i},counts(i))  
end
```

Unknown: 1544 counts  
formerly smoked: 885 counts  
never smoked: 1892 counts  
smokes: 789 counts

```
figure(23);  
clf(23);  
hold on  
histogram('Categories',order_of_counts,'BinCounts',counts)  
title("Distribution of patients and smoking habits")  
ylabel("Counts")  
hold off
```



```
[smoking_col_stroke_pos, smoking_col_stroke_neg] = bin_conditional_extract(strokedata,"smoking_status",strokedata.stroke_pos,stroke_neg);  
[counts2, order_of_counts2] = histcounts(categorical(smoking_col_stroke_pos),smoking_array_entries);
```

```
[counts3, order_of_counts3] = histcounts(categorical(smoking_col_stroke_neg),smoking_array_entries,order_of_counts3)
```

```
fprintf("There were %d participants who formerly or currently smoke and %d of them also suffered a stroke.\n",
```

There were 1674 participants who formerly or currently smoke and 112 of them also suffered a stroke.

```
fprintf("This means that %.2f%% of the participants who have formerly or currently smoked also suffered a stroke.\n",  
        (counts2(2) + counts2(4))/counts2(1:4))
```

This means that 6.69% of the participants who have formerly or currently smoked also suffered a stroke.

```
fprintf("There were %d participants who never smoked and %d of them also suffered a stroke.\n",
```

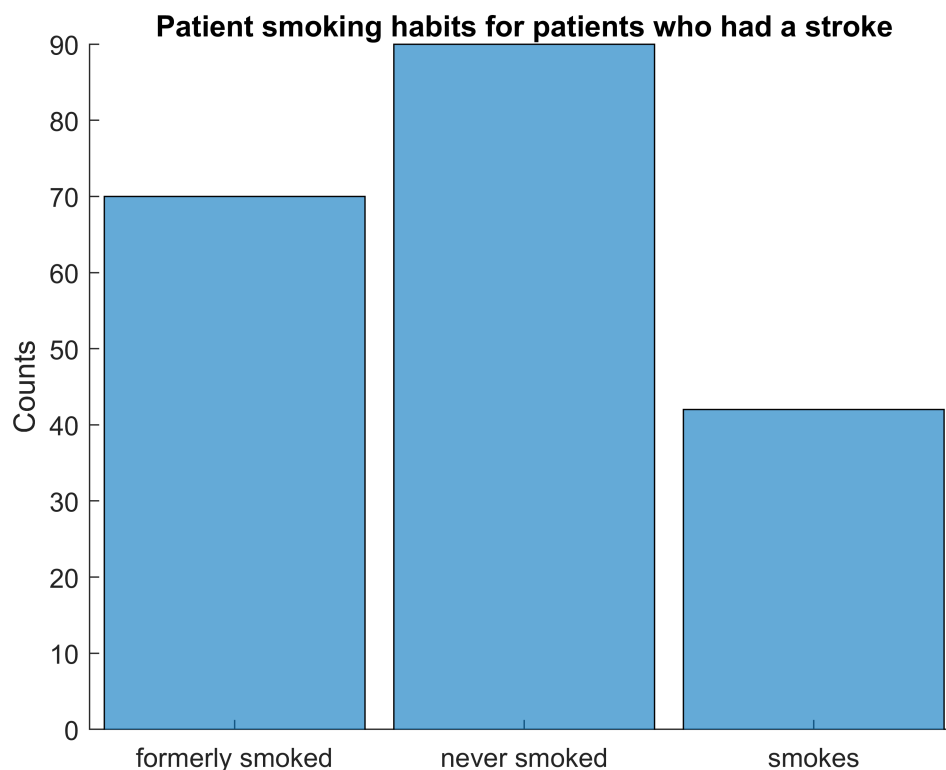
There were 1892 participants who never smoked and 90 of them also suffered a stroke./n

```
fprintf("This means that %.2f%% of the participants who have never smoked also suffered a stroke.\n",
```

This means that 4.76% of the participants who have never smoked also suffered a stroke./n

%Now, here is the information plotted

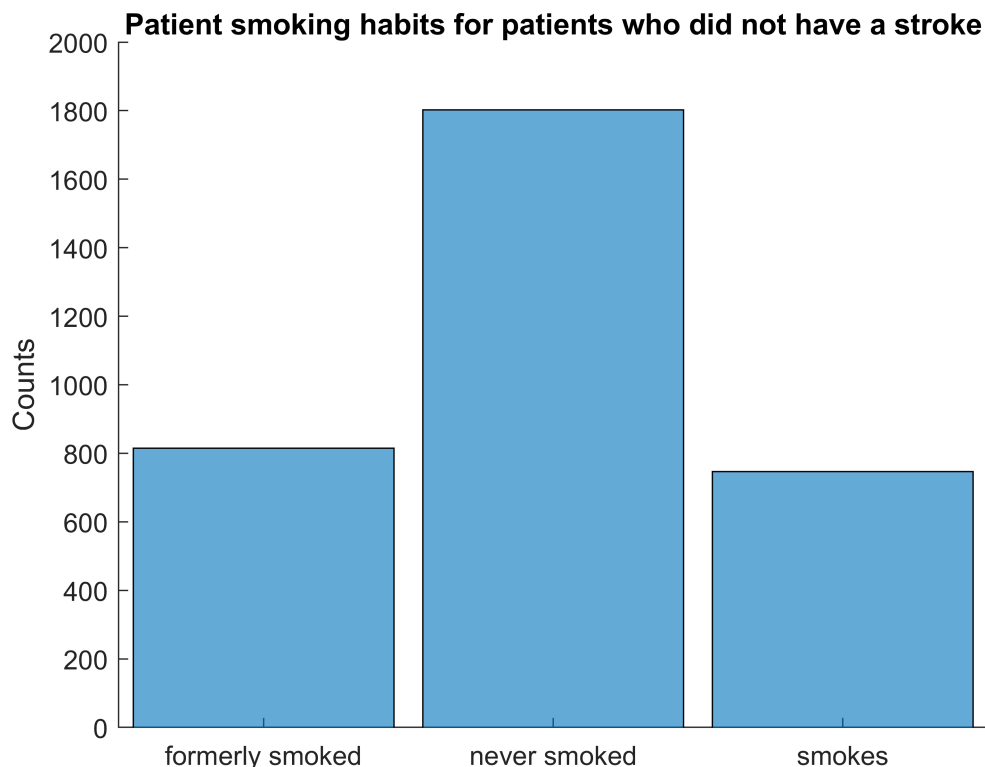
```
smoking_and_stroke_1 = figure(24);  
%smoking_and_stroke_subplot.Position = [800 500 560 420] --- Ignore this  
clf(24);  
hold on  
histogram('Categories',order_of_counts2(2:4),'BinCounts',counts2(2:4))  
title("Patient smoking habits for patients who had a stroke")  
ylabel("Counts")  
hold off
```



```

smoking_and_stroke_2 = figure(25);
clf(25);
hold on
histogram('Categories',order_of_counts3(2:4),'BinCounts',counts3(2:4))
title("Patient smoking habits for patients who did not have a stroke")
ylabel("Counts")
hold off

```



```

formatted_print = "Among stroke patients, %d patients of %d total are smokers (0.2f%%).\n\nAmong\n"
fprintf(formatted_print,counts2(2) + counts2(4), sum(counts2), sum(counts2([2,3]))/sum(counts2)*100)

```

Among stroke patients, 112 patients of 249 total are smokers (64.26%).

Among non-stroke patients, 1562 patients of 4861 total are non-smokers (53.84%).

Note: These numbers include those patients who reported as being 'unknown'.

```

clearvars smoking_array_entries percent_heart_disease_stroke title_subplot2 titleforsubplot2 ti
clearvars order_of_counts order_of_counts2 order_of_counts3 smoking_col_from_table smoking_col_

```

An initial observation about this data is the large number of unknown entries among the patients. More than 20% of the participants are listed as "unknown," which makes it challenging to use this attribute of the data to make predictions about a person's chance of stroke.

Ideally, the data could be used without any of these subjects. However, the smoking patients available are already a small portion of the data, and their other data is valuable. *While performing some data cleaning before*

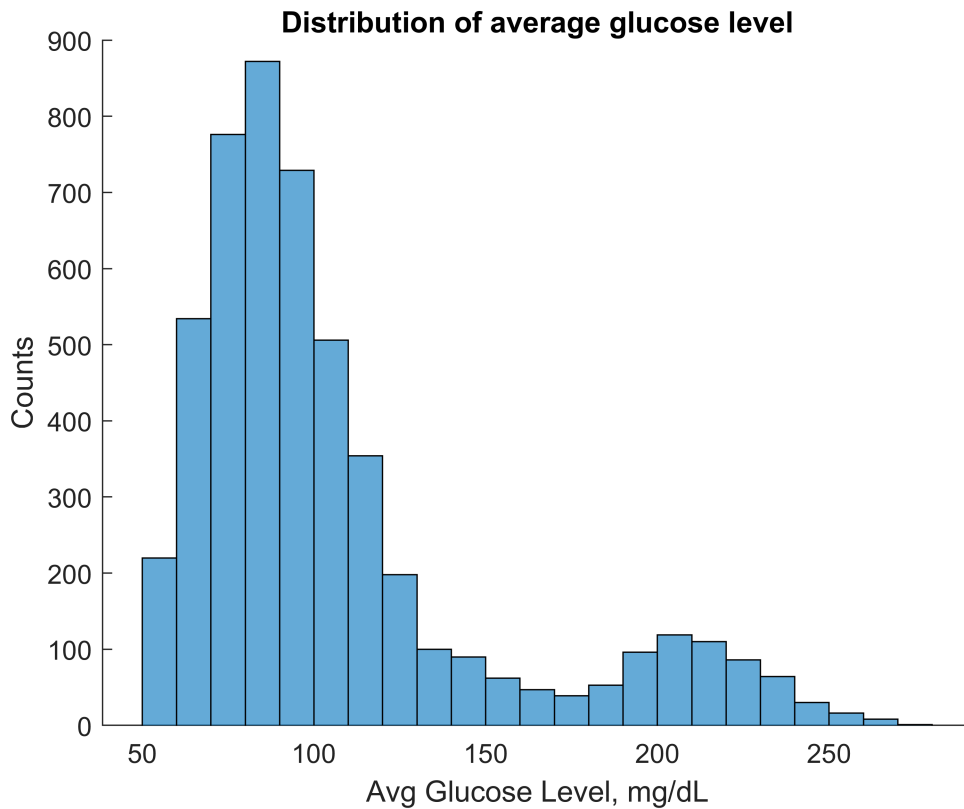
*proceeding to analysis, some of the "unknown" subjects may be removed, but these individuals were generally treated as "never smoking."* This decision errors on the side of indicating that smoking is not a factor that leads to strokes; a conclusion drawn based on the distributions of stroke and non-stroke patients (above), where the proportion of non-smokers is much higher among the patients who have not had a stroke.

Additionally, there does not appear to be a significant variation in smoking patterns in the distributions of stroke patients and non-stroke patients. The relationships of the smokers ("formerly smoked" and "smokes") and non-smokers ("never smoked") in the plot above are not exactly identical, which would indicate no difference at all. The total counts of smokers and non-smokers is approximately equal among the samples of stroke and non-stroke patients. As a model is developed to predict the chance of someone having a stroke, the model will be given this data, and it will be allowed to learn to use this information appropriately even if it turns out that the smoking data does not have a significant impact on its ability to predict that someone is likely to have a stroke.

## Next up, distribution of average glucose level among the study patients

```
f1 = figure(27);
clf(f1);
hold on
histogram(strokedata.avg_glucose_level, 'BinMethod', "auto");

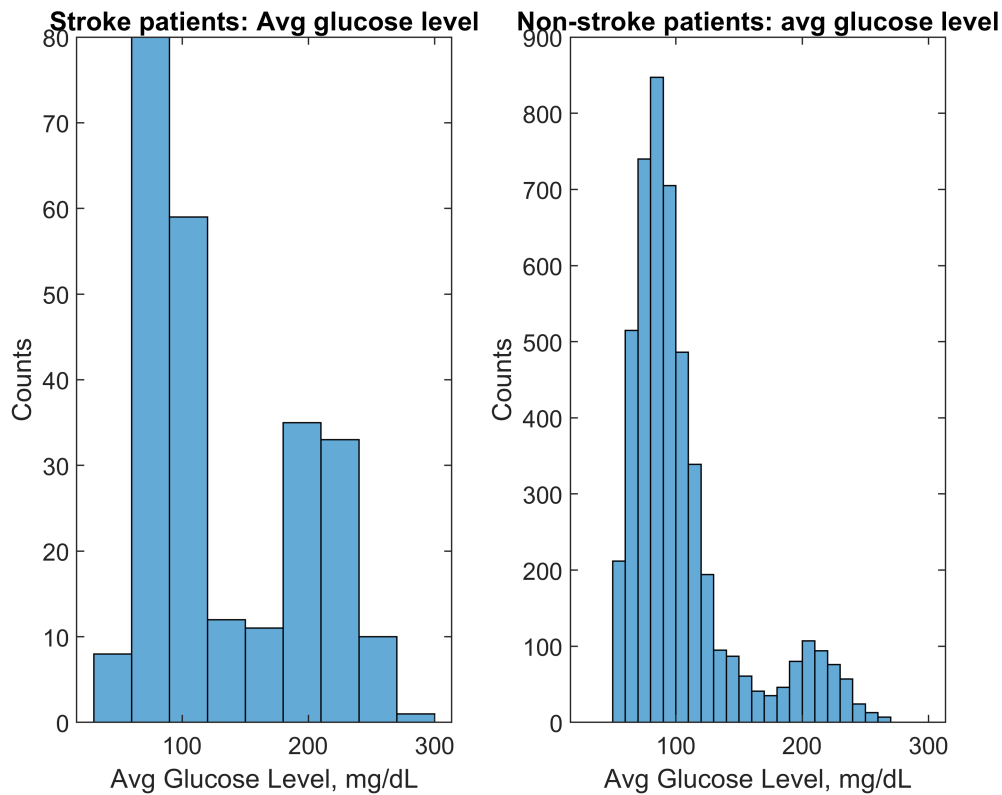
title("Distribution of average glucose level")
xlabel("Avg Glucose Level, mg/dL")
ylabel("Counts")
hold off
```



```
[avg_glucose_stroke, avg_glucose_no_stroke] = bin_conditional_extract(strokedata, "avg_glucose_l
f2 = figure(28);
clf(f2);
hold on

subplot(1,2,1)
h1 = histogram(avg_glucose_stroke, 'BinMethod', "auto");
title("Stroke patients: Avg glucose level")
xlabel("Avg Glucose Level, mg/dL")
ylabel("Counts")

subplot(1,2,2)
h2 = histogram(avg_glucose_no_stroke, 'BinMethod', "auto");
h2.BinLimits = [30, 300];
title("Non-stroke patients: avg glucose level")
xlabel("Avg Glucose Level, mg/dL")
ylabel("Counts")
```



```
clearvars f1 f2 h1 h2 avg_glucose_no_stroke avg_glucose_stroke;
```

From the plots, nothing substantial jumps out about the stroke patients. The distributions have similar shapes, indicating that there isn't a dramatic shift in average glucose level among the stroke or non-stroke patients. This piece of information is a fairly complex piece of data to use because context is important. Glucose levels vary dramatically throughout the day. It was not explicitly stated with the data, but it is presumed that this daily variation is the reason the data was provided as an average (over the course of a day). This addresses one contextual issue, but glucose levels are heavily influenced by a number of health factors, like having diabetes for example. It would take substantial effort and experience to parse out what this distribution of average glucose level might provide about the health of these patients. Instead, the data was provided as is to the classification learner as is, with all of the underlying information included, to allow the high-level statistical model to try to parse out if the information provided by this data is effective in providing information about stroke rates.

## ***Finally, distribution of BMI among the study patients***

BMI is a convenient composite feature to have available in this analysis. It is a representative attribute of a person's height and weight, and plenty of work has been done with the feature already. BMI classification with regards to health has been done as follows:

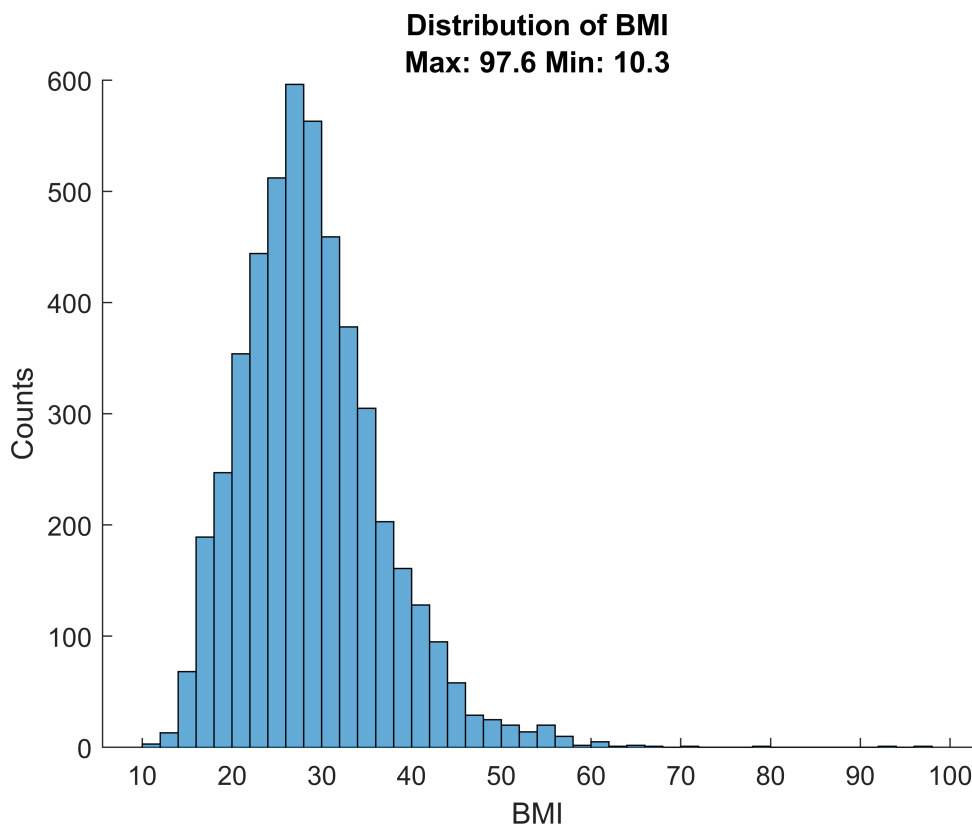
- <18.5 --- "Underweight"
- 18.5 to <25 --- "Normal"



- 25.0 to <30 --- "Overweight"
- >30.0 --- "Obese"

These classifications are helpful context, but the reality is that they can be inaccurate classifications for some. For example, people with large muscle mass, like large athletes, may not be unhealthy but likely fall into the "overweight" or "obese" category. For this reason, the BMI data was used as a value rather than a classification. The classifications might be inaccurate, but the value of BMI itself still provides helpful information about a persons' general body and health that might be useful in identifying strokes.

```
f1 = figure(30);
clf(f1);
hold on
histogram(strokedata.bmi, 'BinMethod', "auto");
titlef1 = sprintf("Distribution of BMI\nMax: %.1f Min: %.1f", max(strokedata.bmi), min(strokedata.bmi));
title(titlef1)
xlabel("BMI")
ylabel("Counts")
hold off
```



```
[bmi_stroke, bmi_no_stroke] = bin_conditional_extract(strokedata, "bmi", "stroke", 1);
f2 = figure(31);
clf(f2);
hold on

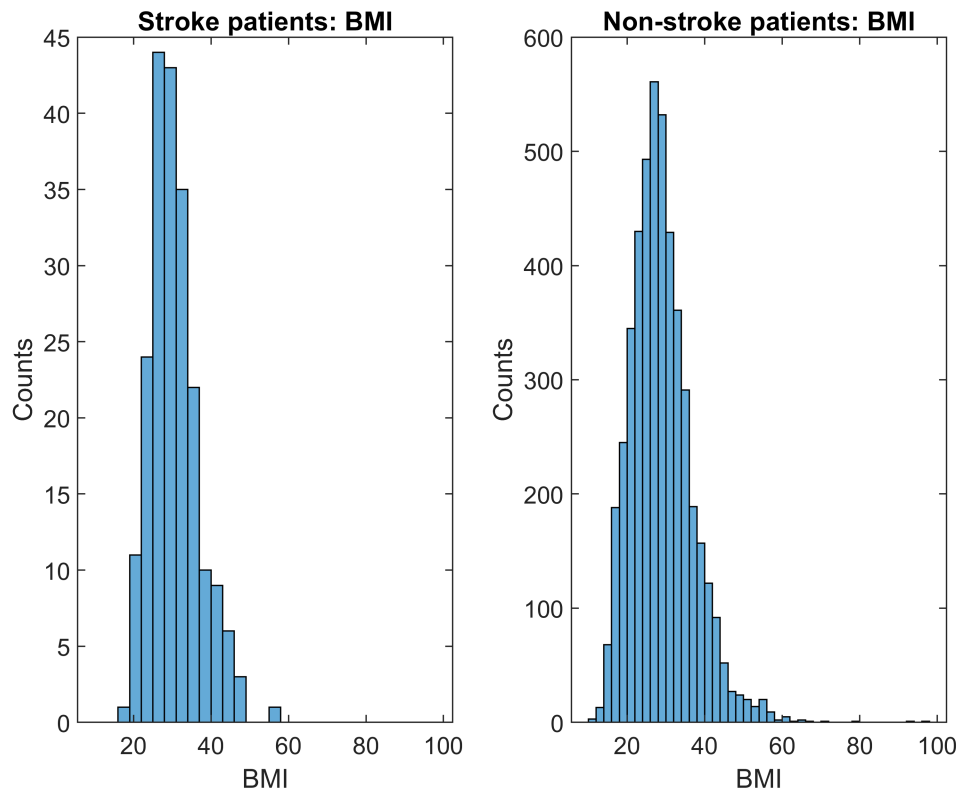
subplot(1,2,1)
```

```

h1 = histogram(bmi_stroke, 'BinMethod', "auto");
h1.BinLimits = [10,98];
title("Stroke patients: BMI")
xlabel("BMI")
ylabel("Counts")

subplot(1,2,2)
h2 = histogram(bmi_no_stroke, 'BinMethod', "auto");
title("Non-stroke patients: BMI")
xlabel("BMI")
ylabel("Counts")

```



```

clearvars f1 titlef1 f2 h2 h1;

```

For the BMI data, there does not appear to be much of a shift in the distribution, which would be expected if any substantial correlation occurred between BMI and stroke rate. These plots do not demonstrate a significant exploration and statistical understanding of the distribution. The data was provided to the classification learning model for further analysis and assessment of significance.

## Section: Data Scrubbing and Preparation

In the Data Analysis section below, some preliminary analysis has indicated that extracting only adults from "strokedata" will yield more accurate information about people who are at risk for strokes. The section of code below will be used to extract a table of only patients above the age of 18 that can be used for the majority of our data analysis.

```
strokedata_18plus = strokedata([strokedata.age >= 18],["gender","age","hypertension","heart_disease","Residence_type","avg_glucose_level","bmi"])  
  
%NOTE: Some variables from "strokedata" were removed.  
% "id" & "ever_married" were not extracted
```

From the preliminary exploration of the data, most attributes provided on the patients in this study did not provide a strong basis to extract the stroke patients. In other words, when separating the stroke patients from non-stroke patients in EDA, very few of the attributes extracted a significant portion of the 249 stroke patients while extracting very few of the non-stroke patients. In particular, three commonly linked risk factors for health complications, (1) heart disease, (2) hypertension, and (3) smoking did not show an obvious percentage increase in the stroke patients compared to the overall patient sample. As noted with hypertension in particular, there might be a slight, and still useful, shift caused by these risk factors. Thus, some feature engineering was done to translate these binary pieces of information ("smoking status" was translated to two groups: (1) smoked at any point and (2) never smoked at all or unknown) to a four point scale that generally represents medical risk factors for the patients in the study.

In the section below, the three features "heart\_disease", "hypertension", and "smoking\_status" were redesigned to fit on a four point scale. The four point scale was added back to the table to replace these three attributes in a new attribute that was named "medical\_risk." A short outline of what was done is provided below before the new feature was created.

- No risk factors (i.e. not a smoker, no heart disease, and no hypertension) === 0
- 1 risk factor === 1
- 2 risk factors === 2
- All risk factors === 3

The model was passed only the "medical\_risk" attribute, not any of the three attributes that contribute to it, with the idea that the group of information is likely a better predictor of stroke than any of the individual attributes. While MATLAB's machine learning model may have been able to come to the same conclusion or may have come to a slightly different conclusion, this approach seemed to be the most tangible and informative all in one.

```
%starting with the data table created above --- strokedata_18plus  
  
%Going to remove "Residence_type" now because it has not been an attribute  
%that we are interested in (will remove by not extracting through really  
  
%Using a function to perform the feature creation and preparation of a new  
%table because I did not want to clutter this script more than necessary  
  
%INPUTS: strokedata_18plus  
strokedata_prepped = datacleanandprep(strokedata_18plus);
```

The new table "strokedata\_prepped" has 7 columns. These columns are listed below.

1. "gender"
2. "age"
3. "Work\_type"
4. "avg\_glucose\_level"
5. "bmi"
6. "stroke"
7. "medical\_risk"

These attributes will be provided to the classification learner in MATLAB in order to develop a model that accurately predicts stroke from this information. The "stroke" column of the table will be the response that the model is trying to classify. It will be using the other 6 features provided as predictors that classify subjects as likely or not to have a stroke.

## **Section: Data Analysis**

The work in this section was dominated by trying to understand and use the classification learner to make stroke predictions from the risk factor data. However, from the exploration of the data, it was clear that age distribution shifted dramatically when looking at stroke patients, so this distribution was analyzed a bit and cleaned up to provide better information for developing a stroke prediction model.

### **Further exploration of age distribution based on other qualifying factors.**

Age is an important qualifier in the context of lifestyle choices effecting a person's chances of having a stroke. It is likely at younger ages that strokes are frequently a result of other medical conditions rather than a child's lifestyle choices, which is what we are generally considering with this dataset. Similarly, elderly subjects in the study are more likely to have additional medical conditions that may play a role in their chances of having a stroke. The next step was to briefly investigate the age distributions of certain attributes (like whether or not someone had a stroke) based on ages.

```
% This line uses "bin_conditional_extract" (read: "Binary Conditional  
% Extract") to extract the ages (in a vector) of people who had a stroke  
[ages_with_stroke, ages_no_stroke] = bin_conditional_extract(strokedata,"age","stroke",1);
```

Now that the ages have been separated into two groups, it is time to look at how ages are distributed for stroke patients compared to non-stroke patients.

```
age_distribution_stroke_conditional = figure(20); %20 is a random figure number
```

```

clf(age_distribution_stroke_conditional)

subplot(1,2,1)
%going to use features of "histogram" function to do binning
ages_stroke_patients = histogram(ages_with_stroke, 'BinMethod', 'auto');

title_subplot1 = sprintf("Ages of stroke patients\nThe bins are groups of %d ages\n(e.g. %d to %d)",
    (ages_stroke_patients.BinEdges(2)-ages_stroke_patients.BinEdges(1)),
    ages_stroke_patients.BinEdges(1), ages_stroke_patients.BinEdges(2));

title(title_subplot1)
xlabel("Ages")

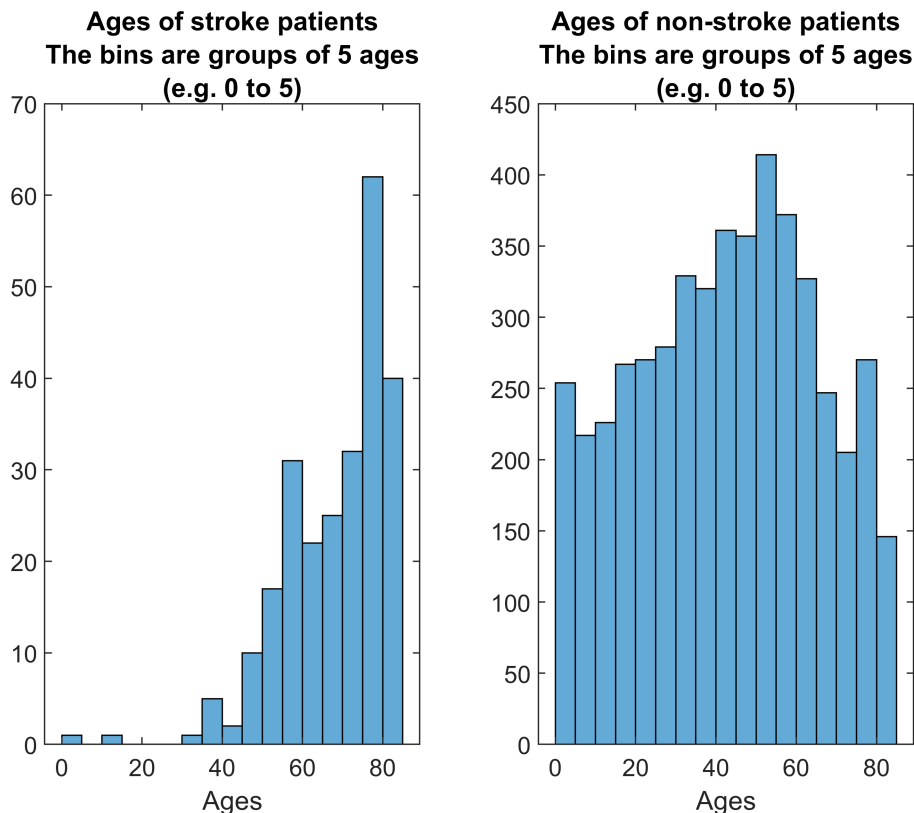
subplot(1,2,2)
ages_no_stroke_pats = histogram(ages_no_stroke, 'BinMethod', 'auto');

title_subplot2 = sprintf("Ages of non-stroke patients\nThe bins are groups of %d ages\n(e.g. %d to %d)",
    (ages_no_stroke_pats.BinEdges(2)-ages_no_stroke_pats.BinEdges(1)),
    ages_no_stroke_pats.BinEdges(1), ages_no_stroke_pats.BinEdges(2));

title(title_subplot2)
xlabel("Ages")

hold off

```



Based on the plots above, it certainly looks like there is a correlation between age and a person's chances of having a stroke. This is useful information with regard to our end goal, understanding what risk factors can be

used to predict a person's chances of having a stroke, and will come up later as we try to find a way to predict a person's chances of having a stroke.

However, most importantly, this age information provides some valuable underlying context to the rest of our data analysis: young children and babies, who do not have useful inputs for many of the attributes investigated (like smoking status, BMI, hypertension, etc.), probably do not contribute helpful information about how these risk factors effect a person's chances of having a stroke. **Moving forward, it will likely be helpful to look at only patients in the age range 18+ because they are full grown adults that are more responsible for their lifestyle choices and accurate information about the relationships between the risk factors and stroke rates. The patients under 18 perhaps suffered a stroke due to other issues unrelated to lifestyle, such as other illnesses or having the genetic tendency for a stroke.**

Before proceeding to remove a set of ages in the data, a test of significance was done on the two distributions above to test our hypothesis that the distribution of ages among stroke patients is significantly different than distribution of all patients in the study.

In other words,

- A two sample t-test was done to determine if the two populations in the plot above represent the same normal distribution of people (Null hypothesis). Alternatively, if there is a significant difference between the distributions, then the test will indicate that there is a significant difference in the data sets, which are solely different based on the condition that one group is former stroke patients.
- Null hypothesis (returns 0 for no significant difference): The two distributions are from the same population of people. In other words, the fact that one group is only stroke patients does not significantly effect the distribution of ages.

```
[h, p] = ttest2(ages_with_stroke, ages_no_stroke);  
fprintf("The t-test returned h = %d. The chance that these distributions represent the same pop
```

The t-test returned h = 1. The chance that these distributions represent the same populations of ages is 0.000%.

The test supports the conclusion that stroke patients have a significantly different age distribution, which can be seen graphically as being higher ages among stroke patients. Now, the stroke data will be re-extracted but will only consider patients who are 18+ in age.

```
clearvars ages_no_stroke ages_with_stroke; %clearing some space first
```

**NOTE:** The 18+ patients were extracted into a new table to use for data analysis in the Data Scrubbing and Extraction section above.

The 18+ patients have been extracted to "**strokedata\_18plus.**"

Now, the age distributions will be displayed briefly for the new data.

```
[ages_stroke, ages_no_stroke] = bin_conditional_extract(strokedata_18plus, "age", "stroke", 1);  
  
age_distribution_stroke_conditional = figure(26); %26 is a random figure number  
clf(age_distribution_stroke_conditional)
```

```

subplot(1,2,1)
%going to use features of "histogram" function to do binning
ages_stroke_patients = histogram(ages_stroke, 'BinMethod', 'auto');

title_subplot1 = sprintf("Ages of 18+ stroke patients\nThe bins are groups of %d ages\n(e.g. %d to %d)",
    (ages_stroke_patients.BinEdges(2)-ages_stroke_patients.BinEdges(1)),
    ages_stroke_patients.BinEdges(1), ages_stroke_patients.BinEdges(2));

title(title_subplot1)
ages_stroke_patients.BinLimits = [18, 84];
xlabel("Ages")

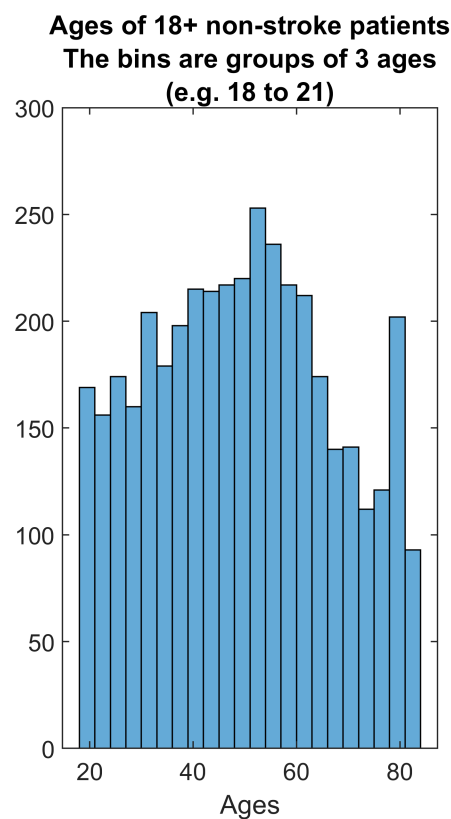
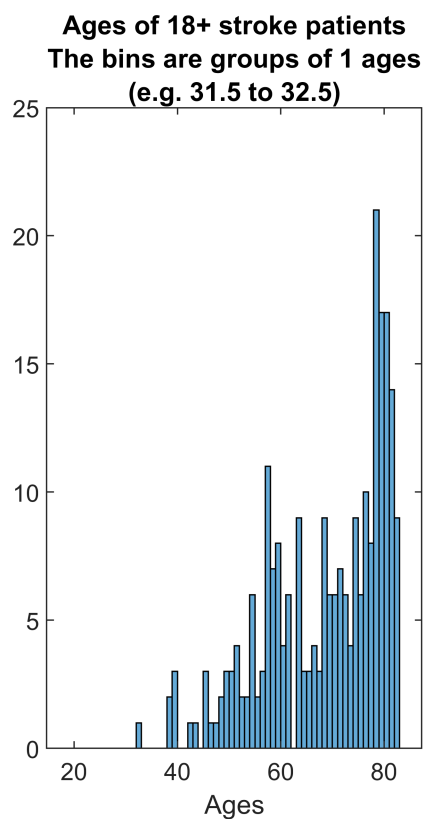
subplot(1,2,2)
ages_no_stroke_pats = histogram(ages_no_stroke, 'BinMethod', 'auto');

title_subplot2 = sprintf("Ages of 18+ non-stroke patients\nThe bins are groups of %d ages\n(e.g. %d to %d)",
    (ages_no_stroke_pats.BinEdges(2)-ages_no_stroke_pats.BinEdges(1)),
    ages_no_stroke_pats.BinEdges(1), ages_no_stroke_pats.BinEdges(2));

title(title_subplot2)
ages_no_stroke_pats.BinLimits = [18, 84];
xlabel("Ages")

hold off

```



```
clearvars h p ages_stroke ages_no_stroke;
```

The data extracted in this section was further modified during feature development, so the data `strokedata_18plus` was not actually used to develop the stroke model - a variation of this data table was used though.

## Data Analysis: Using the Classification Learner to Develop a Stroke Prediction Model

Some scripting and data display is done here, but the remainder of the project (analysis and images of our work) is primarily completed in attached documentation.

Below displays some information about the best prediction model identified (Model 1.1 - Fine Type Tree Class) - named "FineTreeTypeStrokePrediction"

```
load("FineTreeTypeStrokePredictionModel.mat") %should load Model 1.1 using the "Fine Tree" class
fprintf("\n%s\n\n",FineTreeTypeStrokePrediction.About)
```

This struct is a trained model exported from Classification Learner R2020b.

```
fprintf("\n%s\n\n",FineTreeTypeStrokePrediction.HowToPredict)
```

To make predictions on a new table, T, use:

```
yfit = c.predictFcn(T)
replacing 'c' with the name of the variable that is this struct, e.g. 'trainedModel'.
```

The table, T, must contain the variables returned by:

```
c.RequiredVariables
Variable formats (e.g. matrix/vector, datatype) must match the original training data.
Additional variables are ignored.
```

For more information, see How to predict using an exported model.

The section below is meant to demonstrate how predictions can be made with the model. The assessment was done within the classification learner app, so all data has been used for training or assessment by the model, rendering additional predictions on the data as redundant useage. Thus, this demonstration of predictions and comparison to the true results is just to demonstrate how the model is used.

```
randvec = randi([1,4200],1,20);
random_strokedata = strokedata_prepped(randvec,["gender","age","work_type","avg_glucose_level",
predictions = FineTreeTypeStrokePrediction.predictFcn(random_strokedata);

comparisontable = strokedata_prepped(randvec,"stroke");
comparisontable.stroke_modelprediction = predictions;

disp(comparisontable)
```

stroke	stroke_modelprediction
--------	------------------------



0	0
0	0
0	0
0	0
1	0
0	0
0	0
0	0
0	0
0	0
1	0
0	0
0	0
0	0
1	1
0	0
0	0
0	0
0	0
0	0

**It's gonna be a lot of zeros ... which was basically the problem dicussed in much of our analysis.**