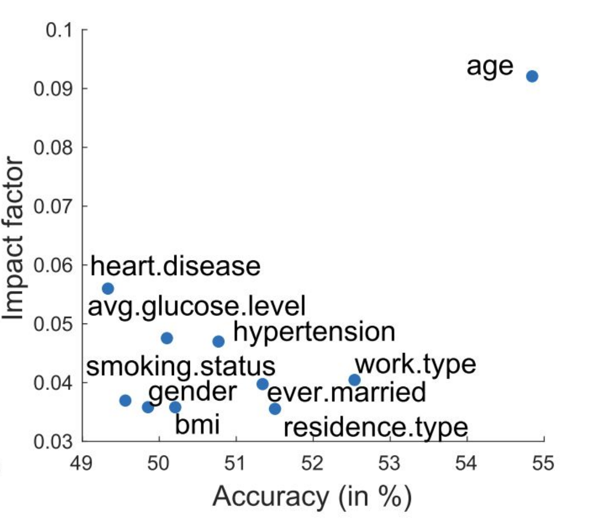
Team SMSU: Accuracy and Bias in Predicting Stroke Occurrence

By: Shaan Malhotra, Uzair Memon, Mohamed Abuelreich, Sathya Rajesh

Objectives

- Analyze literature relevant to our topic
- Recreate stroke occurrence models used in one of the pieces of literature
- Create our own stroke occurrence prediction models
- Compare the accuracy of the two sets of models
- Analyze the bias present in the literature’s models and our own
- Mitigate Bias in our own models
- Re-examine our models

Article #1: “Identifying Stroke Indicators Using Rough Sets” by Pathan et al.

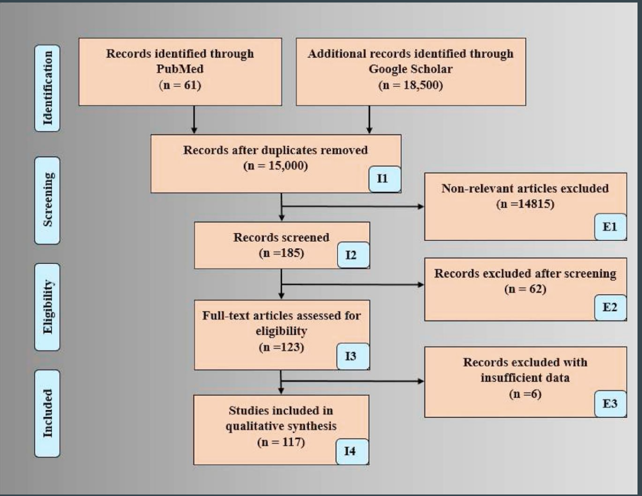


(d) Proposed approach

- Discusses the use of data mining techniques for the accurate prediction of stroke based on risk factors associated with electronic health records (EHRs)
- The choice of feature-selection methods can help improve prediction accuracy and efficient data management of archived input features.
- The article proposes a novel rough-set based technique for ranking the importance of various EHR records in detecting stroke, which can be applied to any dataset comprising binary feature sets.
- Concludes that **age, average glucose level, heart disease, and hypertension** were the most essential attributes for detecting stroke in patients.
- The proposed technique is benchmarked against other popular feature-selection techniques and obtained the best performance in ranking the importance of individual features in detecting stroke.
- Emphasizes the importance of selecting only the most discriminatory

Pathan, Muhammad Salman, et al. "Identifying stroke indicators using rough sets." IEEE Access 8 (2020): 210318-210327.

Article #2: “Understanding the bias in machine learning systems” by Suri et al.



- Assess the difference in risk of bias (RoB) in ML and non-ML cardiovascular disease predictors
- PRISMA model:
  - 117 Studies: 24 pure ML and 14 pure non-ML
  - 3 Bias bins:
    - Low Bias (5 ML vs 3 NML)
    - Medium Bias (10 ML vs 4 NML)
    - High Bias (9 ML vs 7 NML)
  - Common Attributes:
    - smoking (86%), hypertension (76.6%), BMI (70%), family history (63.4%), and ethnicity (38.4%)
- Analytical slope method:
  - bias in the ML papers were significantly less than the bias in the non-ML papers, a roughly 43% difference

Suri, Jasjit S., et al. "Understanding the bias in machine learning systems for cardiovascular disease risk assessment: The first of its kind review." Computers in biology and medicine (2022): 105204.

Article #3: "Predicting Stroke from Electronic Health Records" by Nwosu et al.

- Principal Component Analysis (PCA): provide information for which attributes were responsible for the majority of the variance in the response variable
    - 3 parts: scree plot, biplot, subspace representation
    - Showed that we should use all the attributes for building the models
- Built 3 models for predicting stroke
    - Decision tree, random forest, and neural network

Approach	**Average accuracy (over 1000 experiments)**
Decision Tree	74.31%
Random Forest	74.53%
Neural Network	75.02%

Nwosu, Chidozie Shamrock, et al. "Predicting stroke from electronic health records." 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE, 2019.

Our Dataset

- Sourced from kaggle
  - Originally from an electronic health record provided by McKinsey & Company
- Medical and demographic information for 5,110 patients
- 12 attributes in total
  - 11 input variables
  - 1 response variable (whether or not the patient had a stroke)
- Key observations:
  - More women patients than men patients
  - Only 548 patients had a stroke
  - Major imbalance in classes (will require downsampling)

Variables in our dataset

- Age: double
- Gender: male, female, other
- BMI: double
- Smoking status: either 0 or 1, with formerly smoking being labeled as 1
- Work type: prive, self-employed, government job
- Residence type: urban or rural
- Average glucose levels: double
- Hypertension: yes or no
- Heart disease: yes or no
- Ever married: yes or no

Data cleanup

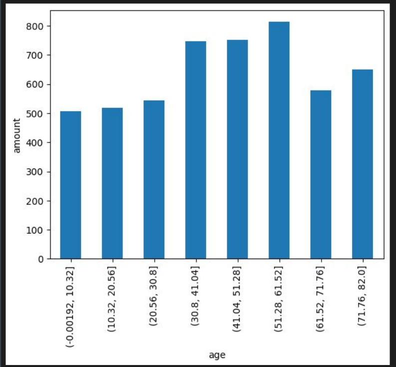
- For this particular dataset, there was only a minimal amount of data cleanup/processing needed
- Smoking status was converted into a binary value
  - They either smoked or didn't, and anyone who was “unknown” was changed to NaN
- Ages were binned
  - 7 bins, 8-10 years in size each

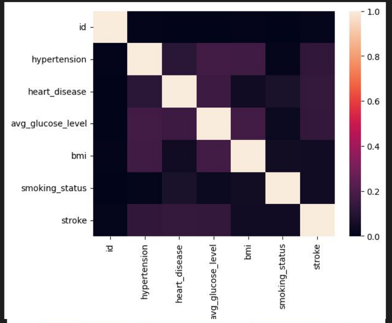
EDA

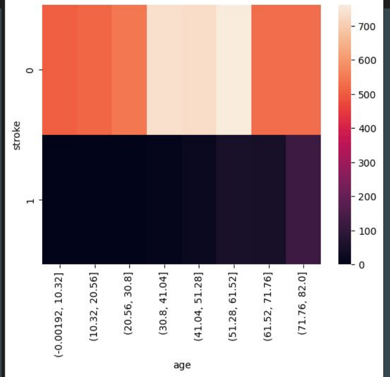
- When looking at the age distribution of the dataset, it is seen that the majority of the people in the dataset are between 30-61

- The strongest factors correlated to having a stroke are hypertension, heart disease, and average glucose levels

- Unsurprisingly, the older people in the dataset had a higher rate of stroke than their younger counterparts.







Models that we made
KNN

```
subset = dataset.groupby('stroke').sample(n=249, random_state=27)
subset.stroke.value_counts()
#taking a subset of the data to have equal number of stoke patients, if this step is not done you'll get
#95% accuracy because the there is an imbalance between people that have stokes and people that don't
#In the 5000 person dataset, only 249 people actually have a stoke.
```

```
0    249
1    249
Name: stroke, dtype: int64
```

```
x_train, x_test, y_train, y_test = train_test_split(subset, labels, random_state=42, test_size=0.2)
#split the data, 80% train and 20% test
```

```
scaler = StandardScaler()
x_train = scaler.fit_transform(x_train)
x_test = scaler.transform(x_test)
```

```
classifier = KNeighborsClassifier(n_neighbors=11, p=2,metric='euclidean')
classifier.fit(x_train, y_train)
```

```
cm = confusion_matrix(y_test, y_pred)
print (cm)
```

```
[[36 15]
 [12 37]]
```

```
print(accuracy_score(y_test, y_pred))
```

```
0.73
```

Random forest

```
x_train, x_test, y_train, y_test = train_test_split(subset, labels, random_state=0, test_size=0.20)
#80% test, 20% test
```

```
rf = RandomForestClassifier()
rf.fit(x_train, y_train)
```

```
RandomForestClassifier()
```

```
y_pred = rf.predict(x_test)
```

```
cm = confusion_matrix(y_test, y_pred)
print (cm)
```

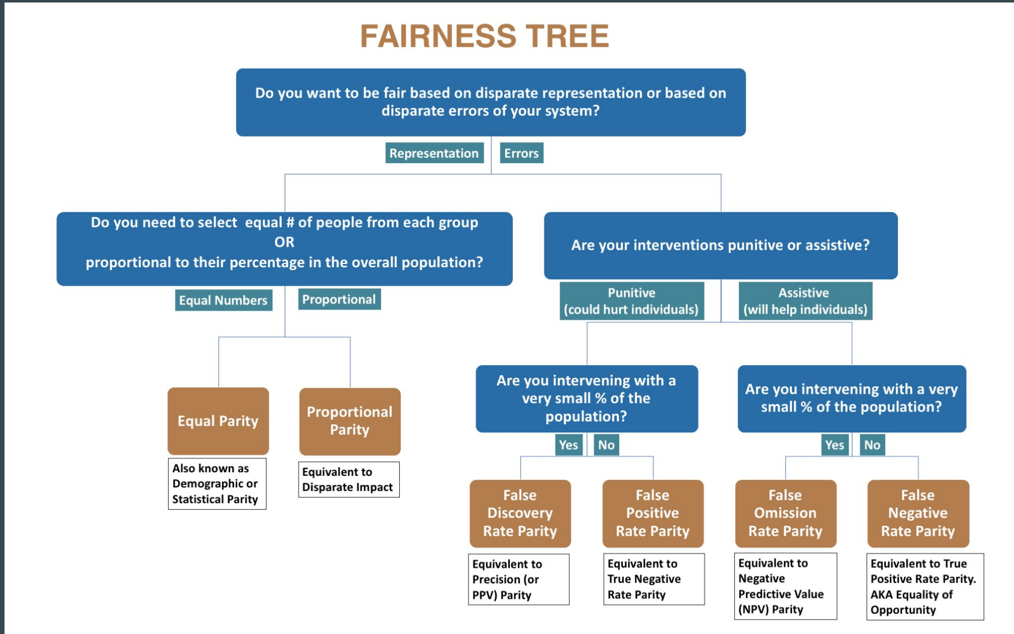
```
[[35 14]
 [ 8 43]]
```

```
print(accuracy_score(y_test, y_pred))
```

```
0.78
```

Analysis of Bias in the Research Article's Models

- Protected attributes we focused on: gender, age, residence type
- Aequitas tool:
    - Reference Groups (majority):
        - Gender: Female
        - Age: 33.50-54.00
        - Residence type: Urban
    - Disparity Intolerance: 80%
    - Tests we used:
        - Equal Parity
        - Proportional Parity
        - False Positive Rate Parity
        - False Omission Rate Parity
        - False Negative Rate Parity



http://aequitas.dssg.io/

Results

- Neural Network:
  - All tests failed

Audit Results: Summary

Equal Parity - Ensure all protected groups are have equal representation in the selected set.	Failed	Details
Proportional Parity - Ensure all protected groups are selected proportional to their percentage of the population.	Failed	Details
False Positive Rate Parity - Ensure all protected groups have the same false positive rates as the reference group).	Failed	Details
False Negative Rate Parity - Ensure all protected groups have the same false negative rates (as the reference group).	Failed	Details
False Omission Rate Parity - Ensure all protected groups have equally proportional false negatives within the non-selected set (compared to the reference group).	Failed	Details

- Random Forest:
  - Only FNRP passed

Audit Results: Summary

Equal Parity - Ensure all protected groups are have equal representation in the selected set.	Failed	Details
Proportional Parity - Ensure all protected groups are selected proportional to their percentage of the population.	Failed	Details
False Positive Rate Parity - Ensure all protected groups have the same false positive rates as the reference group).	Failed	Details
False Negative Rate Parity - Ensure all protected groups have the same false negative rates (as the reference group).	Passed	Details
False Omission Rate Parity - Ensure all protected groups have equally proportional false negatives within the non-selected set (compared to the reference group).	Failed	Details

- Decision Tree:
  - All tests failed

Audit Results: Summary

Equal Parity - Ensure all protected groups are have equal representation in the selected set.	Failed	Details
Proportional Parity - Ensure all protected groups are selected proportional to their percentage of the population.	Failed	Details
False Positive Rate Parity - Ensure all protected groups have the same false positive rates as the reference group).	Failed	Details
False Negative Rate Parity - Ensure all protected groups have the same false negative rates (as the reference group).	Failed	Details
False Omission Rate Parity - Ensure all protected groups have equally proportional false negatives within the non-selected set (compared to the reference group).	Failed	Details

Bias Mitigation

- Used the Aequitas output from the research article models as a starting point
- Tried to address the shortcomings showcased by each of the tests
- Improved random downsampling:
  - Expand to all the attributes that failed the equal and proportional parity test (gender and age)
- Improve data cleaning methods for “strange” outputs
  - Mainly reorganizing the age categories
- Total size of data provided to Aequitas
  - The research models only had viable predictions for 249 rows
  - Changed the training of the model to allow for 498 rows

Analysis of Bias in Our Models

- Before Bias Mitigation:
  - Needed to adjust based on changes made to the format of the data while training

Audit Results: Summary

Equal Parity - Ensure all protected groups are have equal representation in the selected set.	Failed	Details
Proportional Parity - Ensure all protected groups are selected proportional to their percentage of the population.	Failed	Details
False Positive Rate Parity - Ensure all protected groups have the same false positive rates as the reference group).	Passed	Details
False Discovery Rate Parity - Ensure all protected groups have equally proportional false positives within the selected set (compared to the reference group).	Passed	Details
False Omission Rate Parity - Ensure all protected groups have equally proportional false negatives within the non-selected set (compared to the reference group).	Failed	Details

- Random Forest:

Audit Results: Summary

Equal Parity - Ensure all protected groups are have equal representation in the selected set.	Passed	Details
Proportional Parity - Ensure all protected groups are selected proportional to their percentage of the population.	Passed	Details
False Positive Rate Parity - Ensure all protected groups have the same false positive rates as the reference group).	Passed	Details
False Negative Rate Parity - Ensure all protected groups have the same false negative rates (as the reference group).	Passed	Details
False Omission Rate Parity - Ensure all protected groups have equally proportional false negatives within the non-selected set (compared to the reference group).	Passed	Details

- KNN:

Audit Results: Summary

Equal Parity - Ensure all protected groups are have equal representation in the selected set.	Passed	Details
Proportional Parity - Ensure all protected groups are selected proportional to their percentage of the population.	Passed	Details
False Positive Rate Parity - Ensure all protected groups have the same false positive rates as the reference group).	Passed	Details
False Negative Rate Parity - Ensure all protected groups have the same false negative rates (as the reference group).	Passed	Details
False Omission Rate Parity - Ensure all protected groups have equally proportional false negatives within the non-selected set (compared to the reference group).	Passed	Details

Conclusions

- For our project:
    - Accuracy:
        - Research models: Decision tree (74.31%), Random Forest (74.53%), and Neural Network (75.02%)
        - Our models: Random Forest (78%), KNN (73%)
    - Bias:
        - Research models: bias was present for the outlined protected attributes (gander, age, residence type)
        - Our models: bias was initially present but after bias mitigation we greatly reduced any bias present
- For the field in general
    - Machine Learning models in the medical field are only going to expand in terms of abundance and functionality
    - Important to not just understand the biases but also learn strategies to combat them
        - Random downsampling
        - Building off of previously created models after examining their biases
        - Using tools like Aequitas
    - Keeping these strategies in mind when building models can help reduce bias

References

http://aequitas.dssg.io/

Nwosu, Chidozie Shamrock, et al. "Predicting stroke from electronic health records." 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE, 2019.

Pathan, Muhammad Salman, et al. "Identifying stroke indicators using rough sets." IEEE Access 8 (2020): 210318-210327.

Suri, Jasjit S., et al. "Understanding the bias in machine learning systems for cardiovascular disease risk assessment: The first of its kind review." Computers in biology and medicine (2022): 105204.

Thank you for listening!