(Beginning on Slide 7: Data Cleaning and Preprocessing)

Thank you Youxia. Now, we will talk about our data preprocessing steps. First, we drew all our features from NHANES 2015 - 2016 and 2017 - 2018. We removed records where the age of the patient is under 18 or where the patient is pregnant. We then identified features that are best modeled as categorical data. These include the features ‘has diabetes’, ‘current smoking habits’, ‘alcohol consumption in past 12 months’ and ‘physical exercise’. Then we filtered all features by removing any remaining responses that correspond to ‘Missing’, ‘Refused’ or ‘Don’t Know’ leaving us exactly 1136 records remaining. As we can see, the number of total records is lower than we hoped and there is also a size disparity between the two classes. In the next slide, we will address methods to mitigate these issues. And finally, we perform an important step which is normalizing over all the numerical attributes.

(Slide 8: Methodology)

Onto our methodology. Now that we have our clean data, we will split into training, validation, and testing, 70-15-15 respectively. In light of the challenges posed by the data, our goal is to improve the learning of our model. We implemented kernel density estimation to help with that. First we split our training set classwise. So we have one set that is with sleep disorders and another set with no sleep disorders. KDE in our case, will make an estimate of the underlying distribution of the each set and keep the records that most likely fall into the densest areas of the distribution. Then we combine the remaining records from both sets and shuffle. Our hope is that by keeping the top k% of records we can strengthen the observed characteristics for each class, thus improving model learning.

What follows is tuning hyperparameters over the validation set, and then using the best hyperparameters to evaluate over the testing set.

(Slide 9: Logistic Regression)

Our baseline model is logistic regression. As we learned in class, the hyperparameter to tune for is regularization strength, I will call it ‘r strength’. Generally, if ‘r’ strength is too high a logistic regression will underfit. The general trend is observed here, although I will note some nuances with our data. For our dataset, our model performs better when ‘r’ strength is between 0.2 and 0.5. Which means that is prefers not to be penalized too much for model complexity. Using the best hyperparameters lets look at evaluation results. Overall accuracy over 20 iterations is about 67%, but we can clearly see that Logistic Regression performs much better with the majority class, which is sleep disorder positive, than the minority class, which is sleep disorder negative. I’ve also tuned the parameters for KDE and found the best combination would be to keep the top 80% of the negative class and the top 50% of the positive class. These metrics signal room for improvement, which we will believe can happen with addition of more relevant features than what we currently have.

(Slide 10: Random Forest)

Next, we implemented a random forest classifier. For this model we tuned on number of decision trees. We've set the maximum depth of each decision tree to be at most 10, though we generally find that a depth of around 4 - 7 is ideal. Generally, we should find that as the number of trees gets larger the model performs better. While that is true with our model, I have a trendline here because rate of improvement as the number trees grows is actually very small. If we were picking the best hyperparameter based on highest accuracy we find that the number of trees is around 15. However, for this model, I've picked the best hyperparameter based on where the trendline is at it's maximum, which is around 45 decision trees. Looking at the evaluation results, we find the overall accuracy is around 72% over 50 iterations. We see that random forest performs better over the sleep disorder class as compared to the no sleep disorder class. One thing I'd like to note is that I found it very important to correctly pick KDE parameters here. I found that random forest has more overfitting potential than logistic regression when controlling for KDE parameters. So I must keep a more condensed version of the negative class to counteract the tendency to overfit for the postive class. In the future, perhaps experimenting with other node splitting criterions like entropy and log loss could improve our model, but the quality of our feature selection would likely make the best impact. Now I will pass to Youxia to talk about SVM.

(Slide 12: Conclusion)

In conclusion, when evaluating which model is best for our objective given our dataset, we have to ask ourselves what is most important? And we believe that it is most important to accurately diagnosis patients who are positive for sleep disorder. That means we are looking for the model with the best f1 score. So we would pick Random Forest or Logistic Regression. As mentioned previously, the quality and quantity of our data was the most limiting factors in producing good results. To take this study to the next level, we hope to immerse more in the academic literature to improve our feature selection choices.

(Slide 13: Healthcare Significance)

Lastly, the objective of our study was to make a positive impact on healthcare. That includes early detection and prevention of sleep disorders, better personalized treatment and enhanced public and academic awareness of sleep disorder.

(Slide 14)

Thank you! And are there any questions?