IBM Coursera Capstone Project:

*

The Decision Between Driving and Staying Home

The Problem

- In 2018, there were 6,734,000 motor vehicle crashes in 2018 (U.S. Department of Transportation)
- Not all of these ended in death, but crashes can be costly in terms of both property and personal health
- **What if, based on weather and road conditions, we could determine whether it is safe to drive or if we should stay home?**

The Data

- Approximately 20,000 attributes
- Dependent Variable:
  - SEVERITYCODE
- Independent Variables:
  - WEATHER
  - ROADCOND
  - LIGHTCOND
  - ADDRTYPE
  - UNDERINFL
- All variables are categorical, so we need a model that will properly predict the outcomes keeping that consistent

Methodology

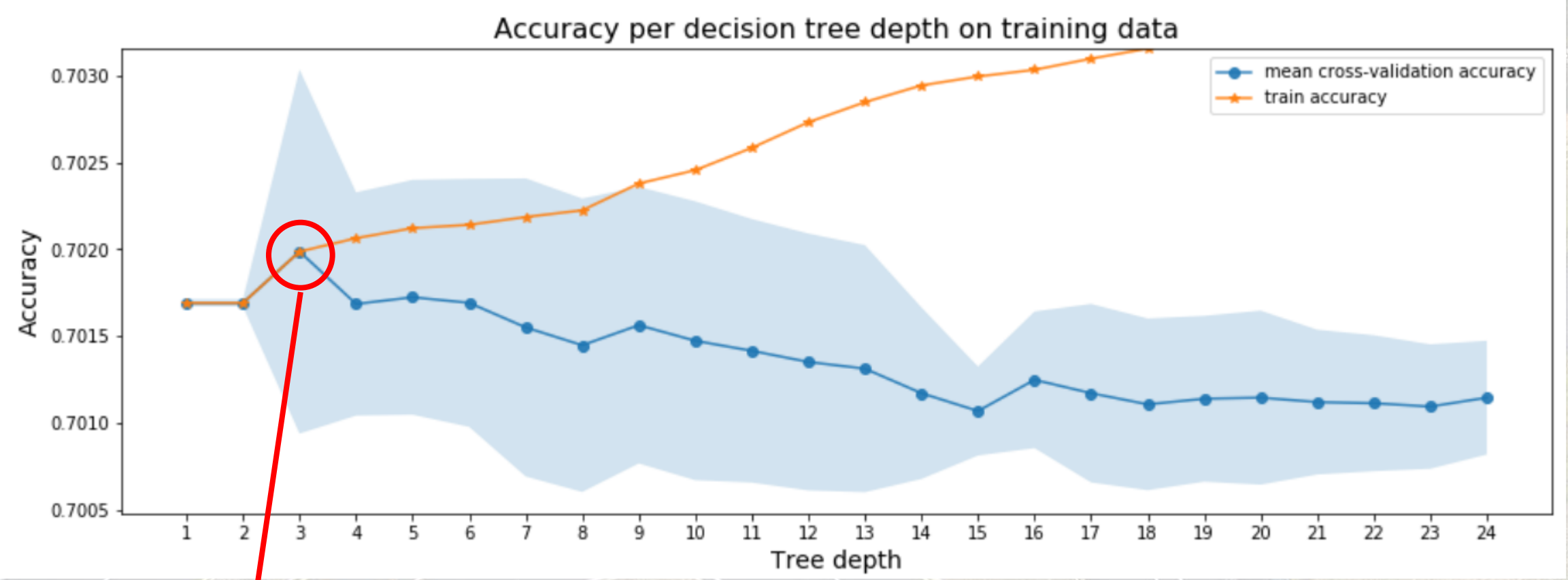
1. Building a new data frame with only the relevant variables
2. Drop the N/A values
3. We are building a model that will help a person determine whether they should stay at home, so we select the decision tree as our machine learning algorithm
4. Convert the categorical variables we have into dummy variables
5. Split the data into training and testing data
6. Build the decision tree and evaluate it

New Data Frame (dropped N/A; pre-dummies)

	SEVERITYCODE	WEATHER	ROADCOND	LIGHTCOND	ADDRTYPE	UNDERINFL
0	2	Overcast	Wet	Daylight	Intersection	N
1	1	Raining	Wet	Dark - Street Lights On	Block	0
2	1	Overcast	Dry	Daylight	Block	0
3	1	Clear	Dry	Daylight	Block	N
4	2	Raining	Wet	Daylight	Intersection	0
...	...	...	...	...	...	...
194668	2	Clear	Dry	Daylight	Block	N
194669	1	Raining	Wet	Daylight	Block	N
194670	2	Clear	Dry	Daylight	Intersection	N
194671	2	Clear	Dry	Dusk	Intersection	N
194672	1	Clear	Wet	Daylight	Block	N

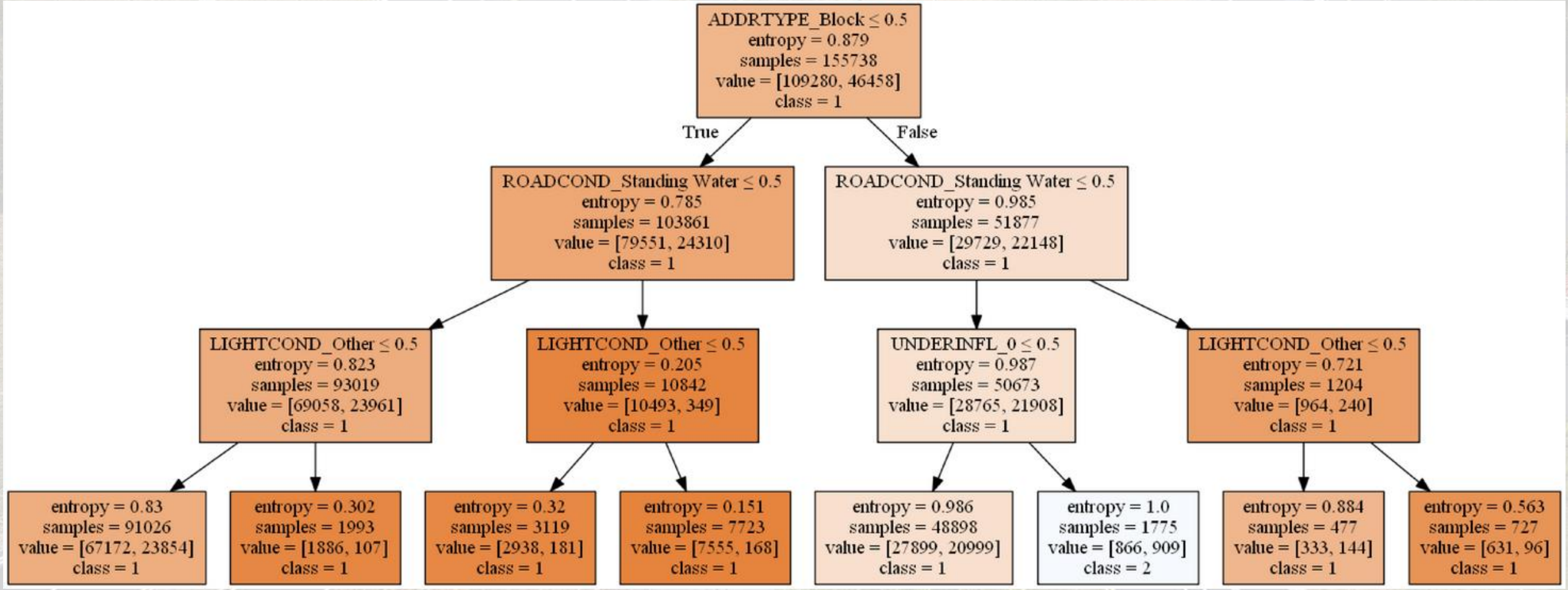
```
SEVERITYCODE      int64
WEATHER          object
ROADCOND         object
LIGHTCOND        object
ADDRTYPE         object
UNDERINFL        object
dtype: object
```

Determine Optimal Decision Tree Depth



The maximum cross-validation accuracy is found at 3, so that is our selected depth

Build Decision Tree



Results & Conclusion

- The decision tree constructed has an accuracy of approximately 70%
- Someone could input the details of their driving situation into the model and predict the possibility of a severe accident
- As with any model, this one has limitations
- Triangulating our decision tree with other algorithms (e.g. a random forest) may ultimately provide drivers with the information they need to prepare to get on the road

Discussion

- The accuracy score of 70% could be improved if :
  - More data were collected
  - More data were made available for integration into model
- We could also include additional conditions (which would require more data):
  - Presence of Hills: make rain/snow more difficult to drive through
  - Population: denser population = higher risk for collisions
  - Gender: correlates between gender and motor vehicle accidents