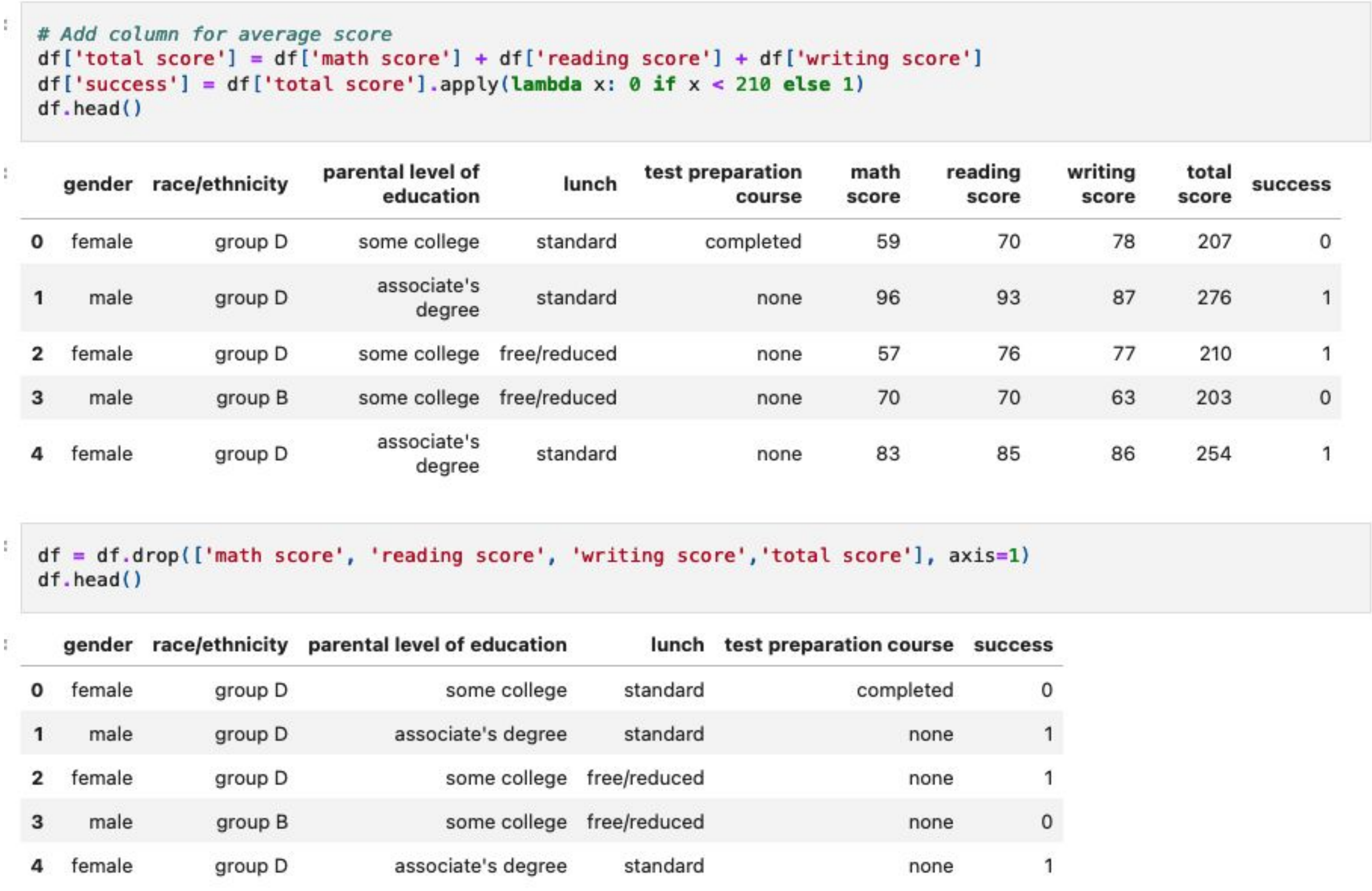
Predictive Modeling in Education

Carol Love
Sahmirah Muhammad
Kendall Sanders
Kelly Schuster-Parades

Kaggle Data



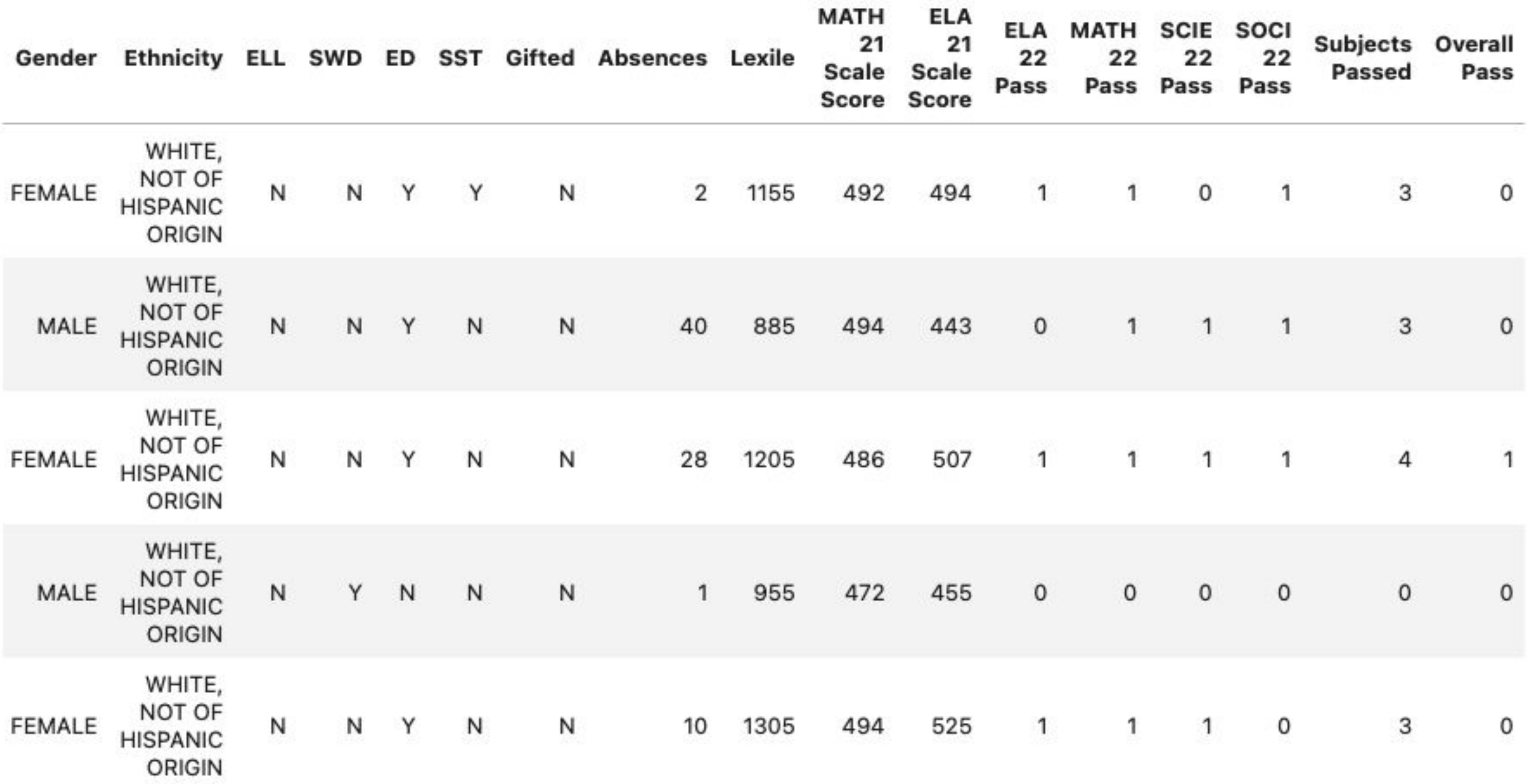
```python
# Add column for average score
df['total score'] = df['math score'] + df['reading score'] + df['writing score']
df['success'] = df['total score'].apply(lambda x: 0 if x < 210 else 1)
df.head()
```

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score	total score	success
0	female	group D	some college	standard	completed	59	70	78	207	0
1	male	group D	associate's degree	standard	none	96	93	87	276	1
2	female	group D	some college	free/reduced	none	57	76	77	210	1
3	male	group B	some college	free/reduced	none	70	70	63	203	0
4	female	group D	associate's degree	standard	none	83	85	86	254	1

```python
df = df.drop(['math score', 'reading score', 'writing score','total score'], axis=1)
df.head()
```

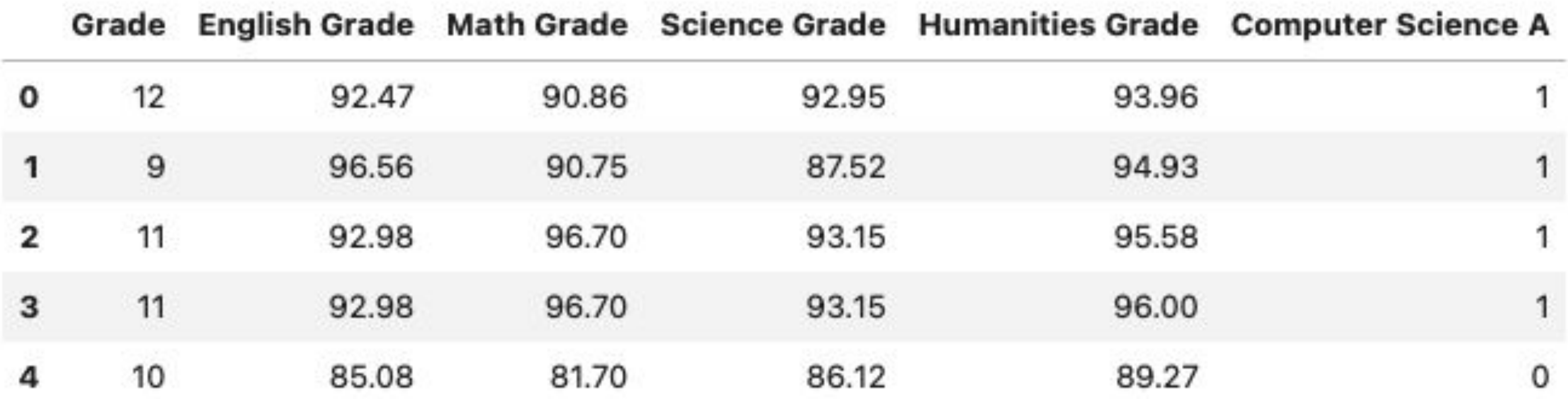
	gender	race/ethnicity	parental level of education	lunch	test preparation course	success
0	female	group D	some college	standard	completed	0
1	male	group D	associate's degree	standard	none	1
2	female	group D	some college	free/reduced	none	1
3	male	group B	some college	free/reduced	none	0
4	female	group D	associate's degree	standard	none	1

Georgia Data



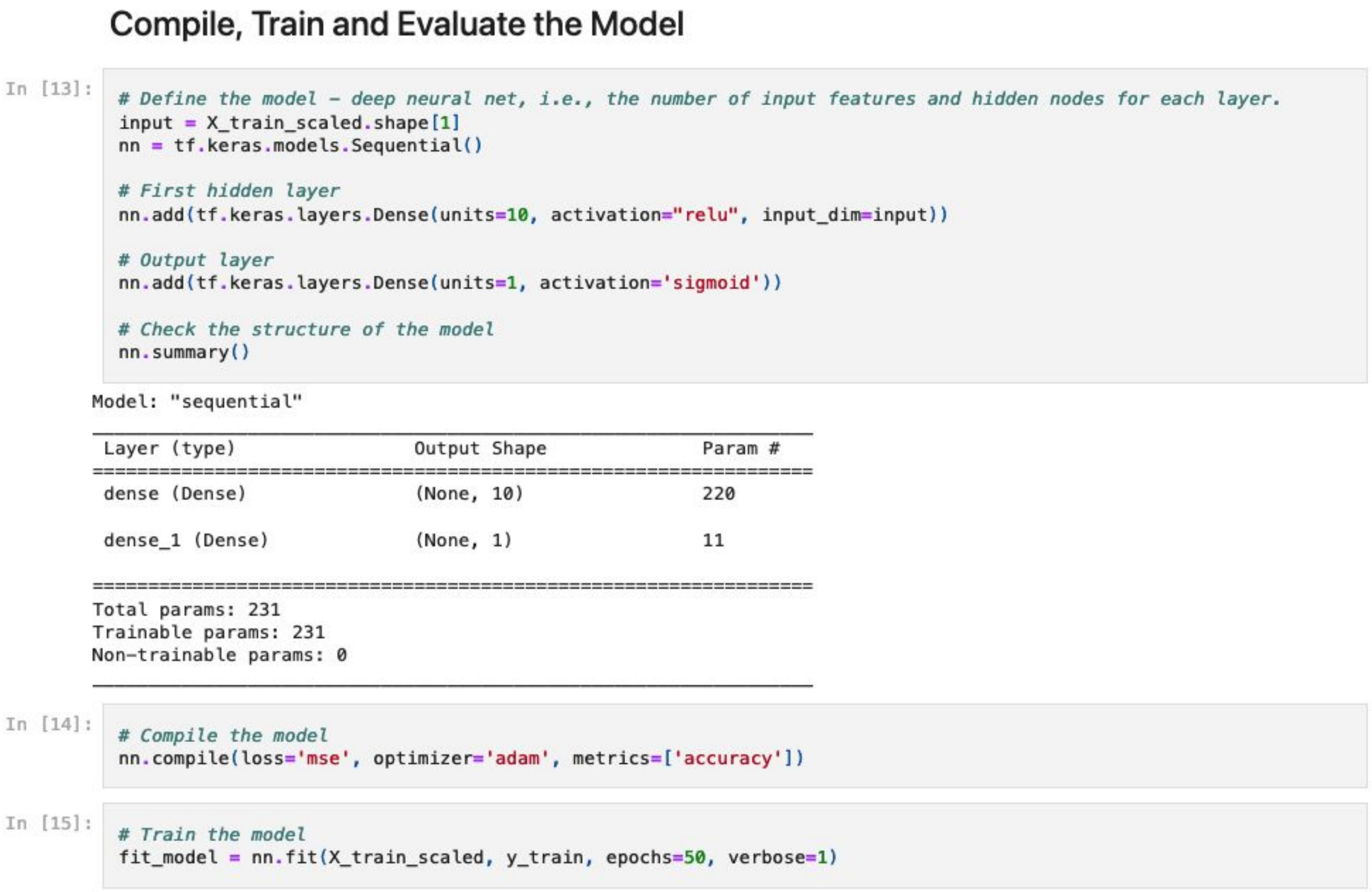
Gender	Ethnicity	ELL	SWD	ED	SST	Gifted	Absences	Lexile	MATH 21 Scale Score	ELA 21 Scale Score	ELA 22 Pass	MATH 22 Pass	SCIE 22 Pass	SOCI 22 Pass	Subjects Passed	Overall Pass
FEMALE	WHITE, NOT OF HISPANIC ORIGIN	N	N	Y	Y	N	2	1155	492	494	1	1	0	1	3	0
MALE	WHITE, NOT OF HISPANIC ORIGIN	N	N	Y	N	N	40	885	494	443	0	1	1	1	3	0
FEMALE	WHITE, NOT OF HISPANIC ORIGIN	N	N	Y	N	N	28	1205	486	507	1	1	1	1	4	1
MALE	WHITE, NOT OF HISPANIC ORIGIN	N	Y	N	N	N	1	955	472	455	0	0	0	0	0	0
FEMALE	WHITE, NOT OF HISPANIC ORIGIN	N	N	Y	N	N	10	1305	494	525	1	1	1	0	3	0

Florida Data



	Grade	English Grade	Math Grade	Science Grade	Humanities Grade	Computer Science A
0	12	92.47	90.86	92.95	93.96	1
1	9	96.56	90.75	87.52	94.93	1
2	11	92.98	96.70	93.15	95.58	1
3	11	92.98	96.70	93.15	96.00	1
4	10	85.08	81.70	86.12	89.27	0

Kaggle Model



Compile, Train and Evaluate the Model

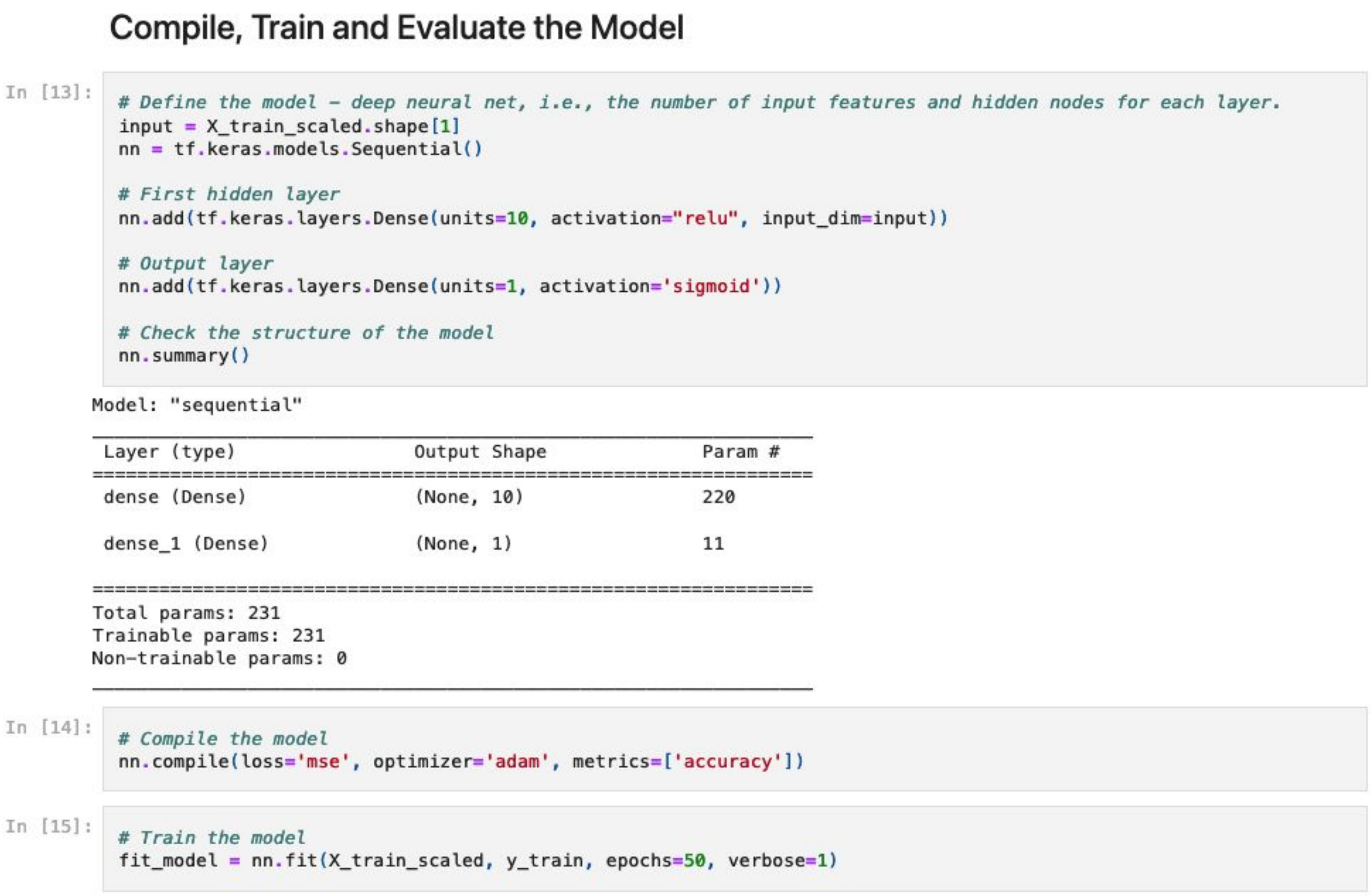
```
In [13]:
# Define the model - deep neural net, i.e., the number of input features and hidden nodes for each layer.
input = X_train_scaled.shape[1]
nn = tf.keras.models.Sequential()

# First hidden layer
nn.add(tf.keras.layers.Dense(units=10, activation="relu", input_dim=input))

# Output layer
nn.add(tf.keras.layers.Dense(units=1, activation='sigmoid'))

# Check the structure of the model
nn.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense (Dense)               (None, 10)                220

 dense_1 (Dense)             (None, 1)                 11

=================================================================
Total params: 231
Trainable params: 231
Non-trainable params: 0
_________________________________________________________________
```

```
In [14]:
# Compile the model
nn.compile(loss='mse', optimizer='adam', metrics=['accuracy'])
```

```
In [15]:
# Train the model
fit_model = nn.fit(X_train_scaled, y_train, epochs=50, verbose=1)
```

Georgia Model



Compile, Train and Evaluate the Model

In [13]:

```
# Define the model - deep neural net, i.e., the number of input features and hidden nodes for each layer.
input = X_train_scaled.shape[1]
nn = tf.keras.models.Sequential()

# First hidden layer
nn.add(tf.keras.layers.Dense(units=10, activation="relu", input_dim=input))

# Output layer
nn.add(tf.keras.layers.Dense(units=1, activation='sigmoid'))

# Check the structure of the model
nn.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense (Dense)               (None, 10)                220

 dense_1 (Dense)             (None, 1)                 11

=================================================================
Total params: 231
Trainable params: 231
Non-trainable params: 0
_________________________________________________________________
```

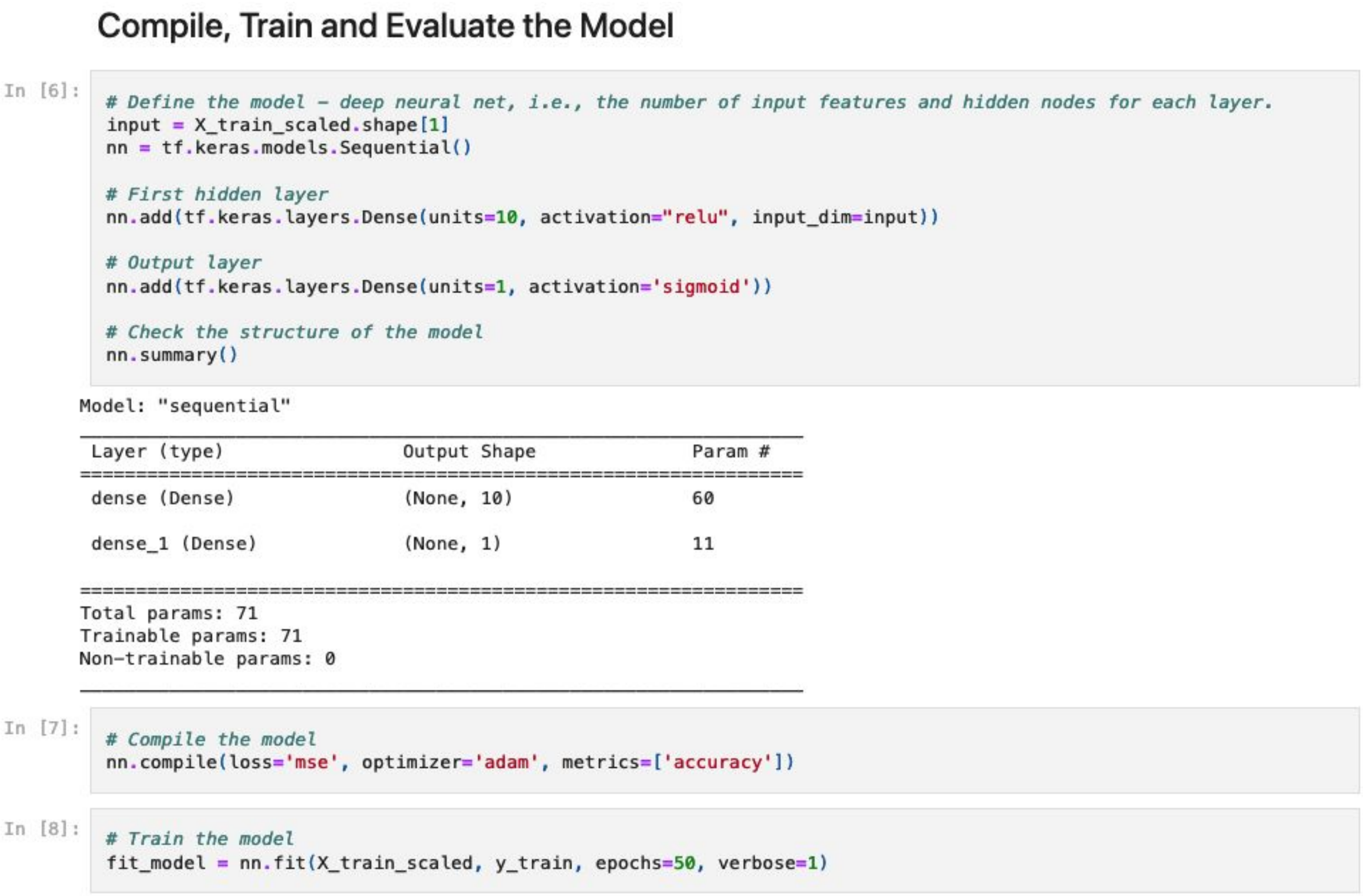
In [14]:

```
# Compile the model
nn.compile(loss='mse', optimizer='adam', metrics=['accuracy'])
```

In [15]:

```
# Train the model
fit_model = nn.fit(X_train_scaled, y_train, epochs=50, verbose=1)
```

Florida Model



Compile, Train and Evaluate the Model

In [6]:

```
# Define the model - deep neural net, i.e., the number of input features and hidden nodes for each layer.
input = X_train_scaled.shape[1]
nn = tf.keras.models.Sequential()

# First hidden layer
nn.add(tf.keras.layers.Dense(units=10, activation="relu", input_dim=input))

# Output layer
nn.add(tf.keras.layers.Dense(units=1, activation='sigmoid'))

# Check the structure of the model
nn.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense (Dense)               (None, 10)                60

 dense_1 (Dense)             (None, 1)                 11

=================================================================
Total params: 71
Trainable params: 71
Non-trainable params: 0
_________________________________________________________________
```

In [7]:

```
# Compile the model
nn.compile(loss='mse', optimizer='adam', metrics=['accuracy'])
```

In [8]:

```
# Train the model
fit_model = nn.fit(X_train_scaled, y_train, epochs=50, verbose=1)
```

Model Accuracy

Kaggle Data Set:

- Reading: 0.616
- Writing: 0.672
- Math: 0.608

- Overall (average >= 70): 0.640

Georgia Milestones Data:

- English Language Arts (ELA): 0.867
- Math (MATH): 0.933
- Science (SCIE): 0.667
- Social Studies (SOCI): 0.900

- Overall (pass all four tests): 0.833

Florida Computer Science Data:

- Computer Science: 0.829