Data for Good: Connecting Graduate Coursework with Real-World Solutions for Nonprofits

SecondHand Hounds

Vasudha Gulati
Shalini Sarathy Komandur Chakravarthy
Andrei Jula

2023 Great Lakes Data & Analytics Summit

OAKLAND UNIVERSITY

Shalini Sarathy Komandur Chakravarthy

Data Flow / Preprocessing
PowerBI - Application Charts / KPIs / Metrics / Insights

SHH Organizational Sponsors & Oakland University Team

- **Carrie Openshaw**, Program Director – “Oversees Program and Outreach Departments as well as shelter partner relationships”
- **Maggie Schmitz**, Director of Marketing – “Oversees Marketing Department, manages SHH social media, managers media / PR inquiries and marketing requests”
- **Tashina Schumacher**, Chief of Staff – “Oversees Operations, Human Resources & Finance”

Left to Right:

Shalini Sarathy Komandur Chakravarthy
James Balwinski
Dr. Vijayan Sugumaran
Phat Pham
Zahra Khan
Andrei Jula
Vasudha Gulati

Animal Rescue as a Nonprofit Organization

- The **SecondHand Hound** – a nonprofit animal rescue organization, founded in 2009
- The organization mainly focuses on finding a home for pets that can no longer be provided care for by their owners, or rescues from kill shelters, in the Midwest area
- During the process, the volunteers provide a temporary safe shelter, veterinary care, rehabilitation or hospice care
- All the involved project activities related to adoption and neonatal programs (pets born with health issues or disabilities), are coordinated by a dedicated and inspired staff
- Some of the activities are collecting donations, creating awareness, identifying and reaching out to possible foster to adopting clients, organizing fostering and volunteering events.

How it all started..

- Bringing their data together
- Simple intake and outcome excel report.
- Find a permanent home for rescued animals as soon as possible.

Requirement and Outcome

LIVE INTAKE JAN				
CANINES	Adult	<5mo	unkown	Subtotal
Stray	1			1
OS	128	39		167
OSeuth				0
Transfer Instate				0
Transfer Outstate				0
Transfer International				0
Seized				0
Other		34		34
FELINES	Adult	<5mo	unkown	Subtotal
Stray	2			2
OS	33	16		49
OSeuth				0
Transfer Instate	5	11		16
Transfer Outstate				0
Transfer International				0
Seized				0
Other				0

LIVE OUTCOMES JAN				
CANINES	Adult	<5mo	unkown	Subtotal
Adoption	74	50		124
Returned to Owner				0
Transfered in State				0
Transferred Out of State				0
Transferred International				0
Returned to Field				0
Other				0
FELINES	Adult	<5mo	unkown	Subtotal
Adoption	14	14		28
Returned to Owner				0
Transfered in State				0
Transferred Out of State				0
Transferred International				0
Returned to Field				0
Other				0

Year

Select all | 2017 | 2018 | 2019 | 2020 | 2021 | 2022

Month

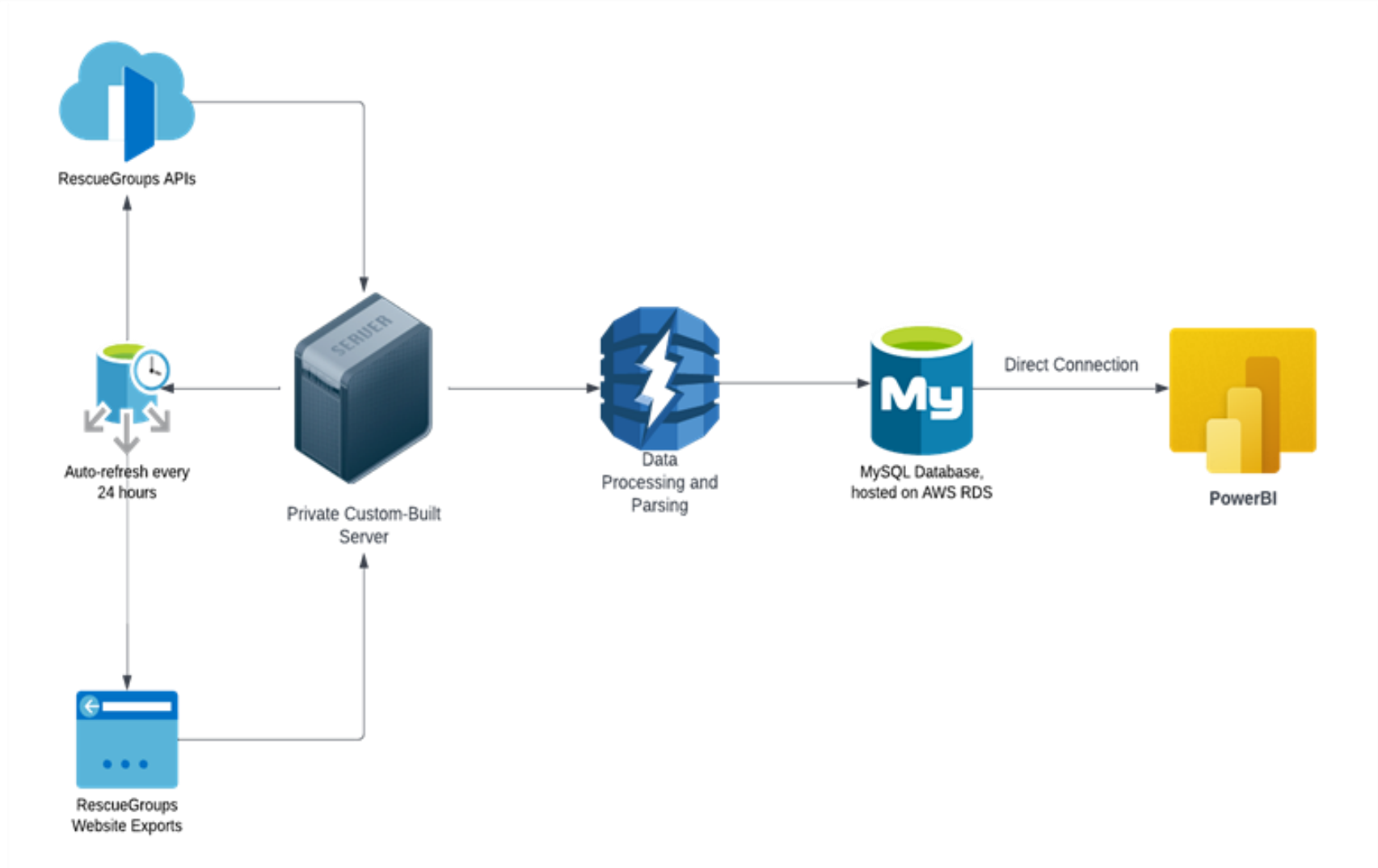
Select all | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10

Year	Month	Species	Stray at Large Adult	Stray at Large upto 5 Months	Stray at Large Age Unknown	Relinquinshed by Owner Adult	Relinquinshed by Owner Up To 5 Months	Relinquinshed by Ov
2022	1	Canine		1				
2022	1	Feline	1					
2022	2	Canine						
2022	2	Feline	1					
2022	3	Canine						
2022	3	Feline						
2022	4	Canine						
2021	1	Canine						
2021	1	Feline	1					
2021	2	Canine						
2021	2	Feline						
2021	3	Canine						
2021	3	Feline						
2021	4	Canine	1					
2021	4	Feline	1					
2021	5	Canine						
Total			**79**	**352**				

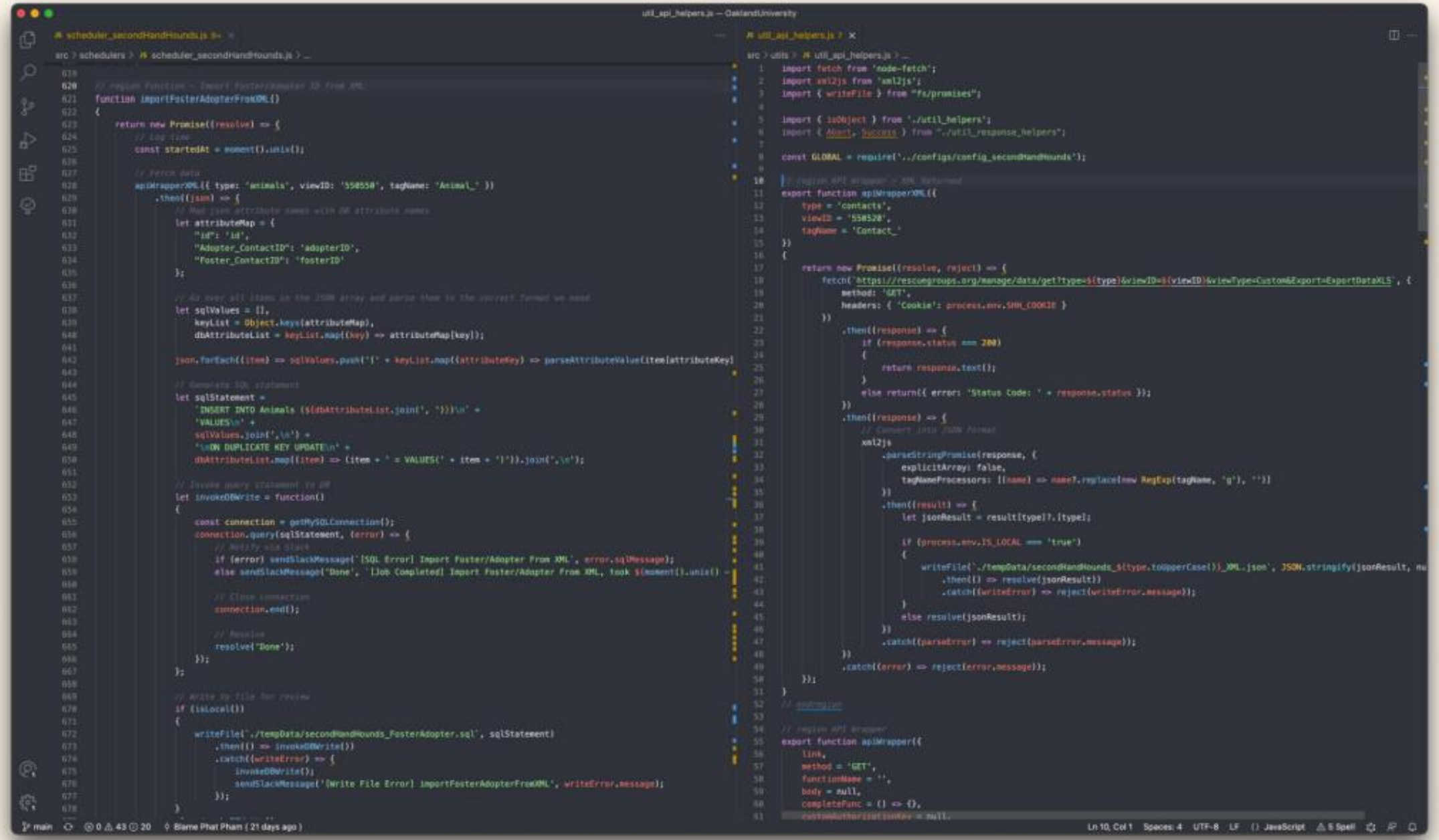
Initial Challenges

1. Data Inconsistencies - Many nulls, headers named differently per different sources for joining, merged information (Ex: multiple pet names in single cell), and many more.
2. Historical Data - Rescue Groups does not hold information prior to updates. (Ex: Who was the last foster if different?)
3. The ETL Process: Gaining access to create an API and converting files
4. Connecting the data to PowerBI (no auto-refresh)
5. Local vs Client (Ex: Python code may work locally but not for client)

ETL Solution Implementation



Portions of the server's code

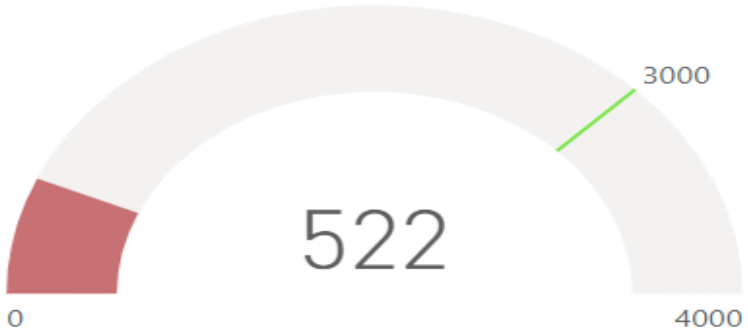


Vasudha Gulati

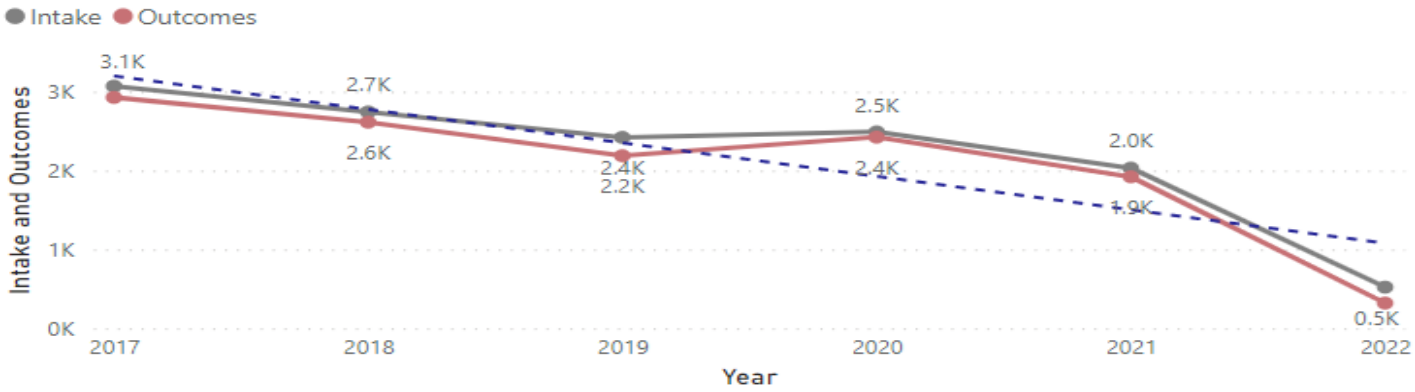
PowerBI - Application Charts / KPIs / Metrics / Insights

Performance Dashboard

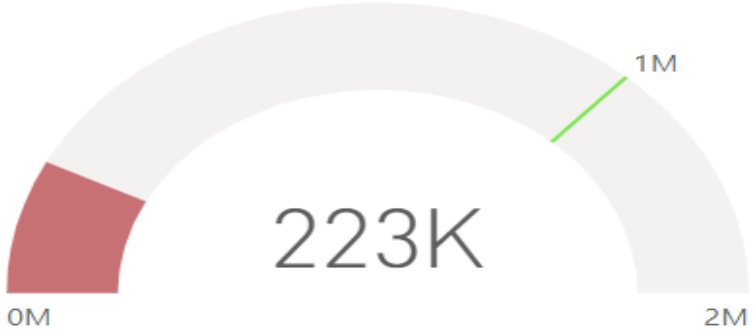
Animals Rescued/Intake 2022



Animal Intake Vs Outcomes Snapshot



Adoption Fees 2022



Animal Adoption Snapshot by Age

127	166	17
Adult Dogs Adopted YTD	Puppies Adopted YTD	Avg Intake Age (m)
14	14	20
Adult Cats Adopted YTD	Kittens Adopted YTD	Avg Outcome Age (m)

Year: Multiple selections | Quarter: All | Month: All | Species: Select all, Cat, Dog

Animal Intake	Animals Adopted	Adoption Fees	Last Intake Date
13.27K	12.39K	5M	Saturday, April 09, 2022

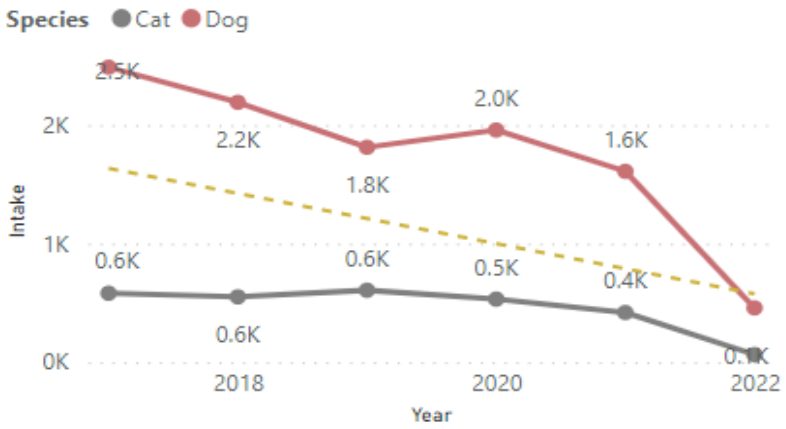
Live Intakes *(by Category and Age Group)*

Category	Adult	Baby (Less < 5 months)	Total
Transfer Outstate	1121	1142	**2263**
Transfer International	121	72	**193**
Transfer Instate	316	368	**684**
Stray	82	349	**431**
Other	2325	1548	**3873**
OS	2659	2631	**5290**
Total	**6739**	**6535**	**13274**

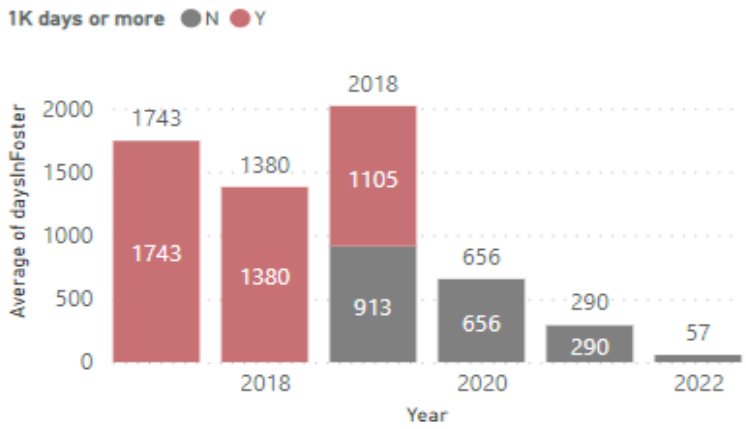
Live Outcomes *(by Category and Age Group)*

Category	Adult	Baby (Less < 5 months)	Total
Transfer Outstate	1114	1056	**2170**
Transfer International	121	61	**182**
Transfer Instate	316	337	**653**
Stray	81	320	**401**
Other	2298	1354	**3652**
OS	2642	2437	**5079**
Total	**6658**	**5734**	**12392**

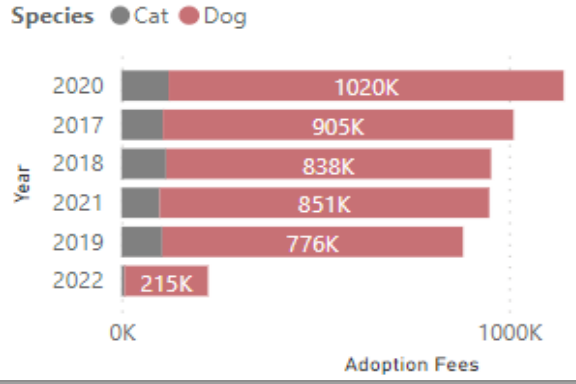
Animals Adopted Yearly Comparison



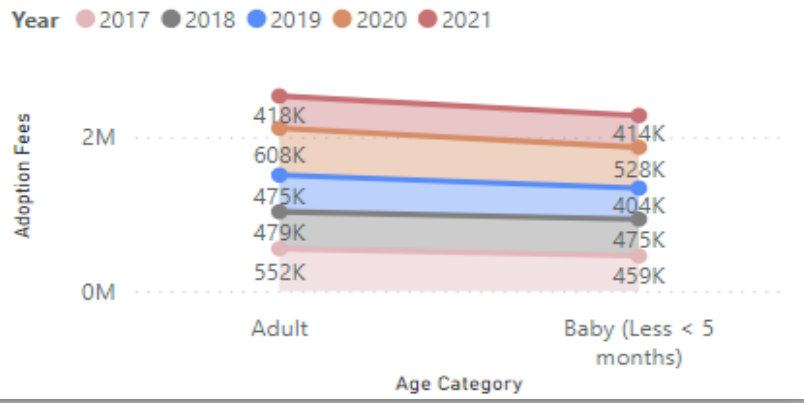
Days in Foster



Adoption Fee Intake YOY



Adoption Fee breakdown by Age Category



Year

Multiple selections

Quarter

All

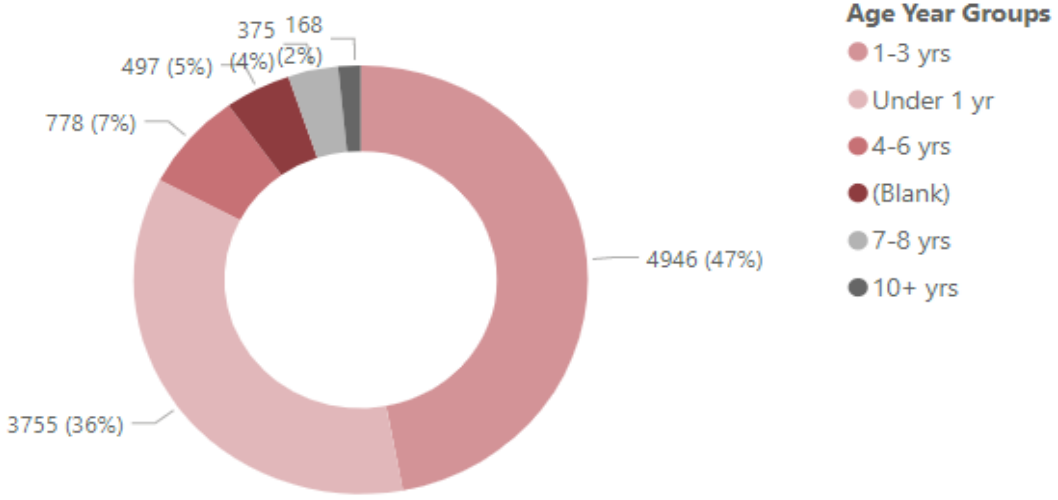
Month

All

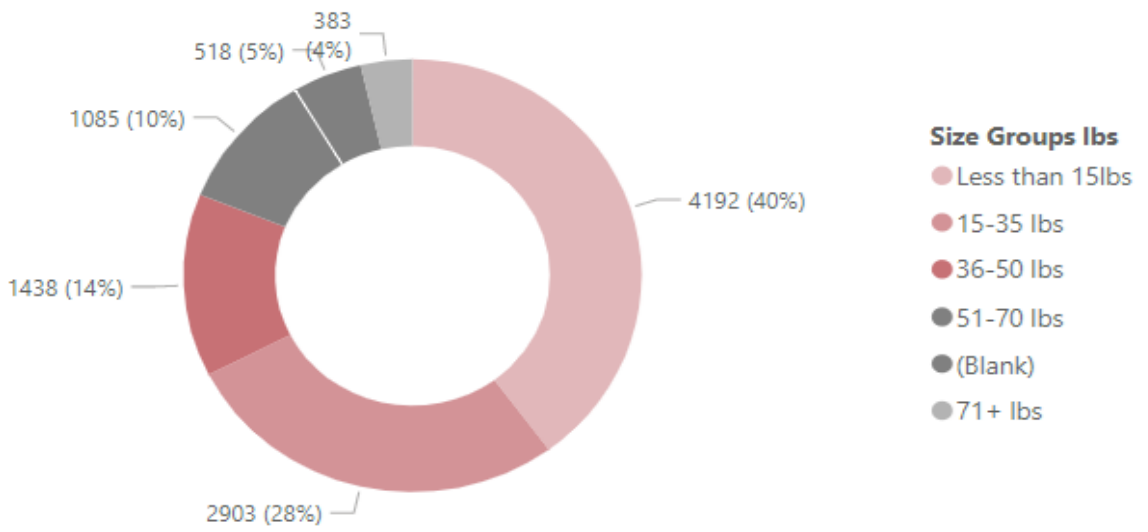
Species

Select all | Cat | Dog

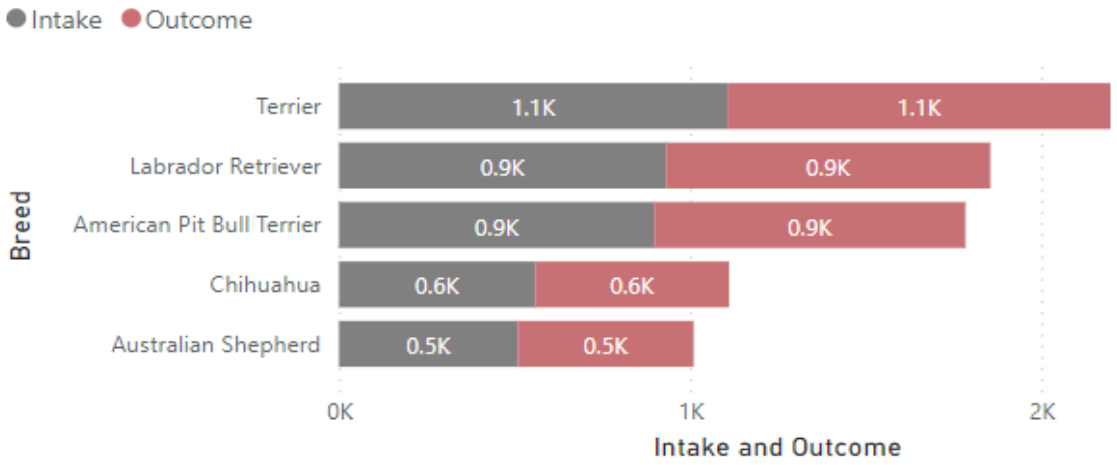
Animal Intake breakdown by Age Groups



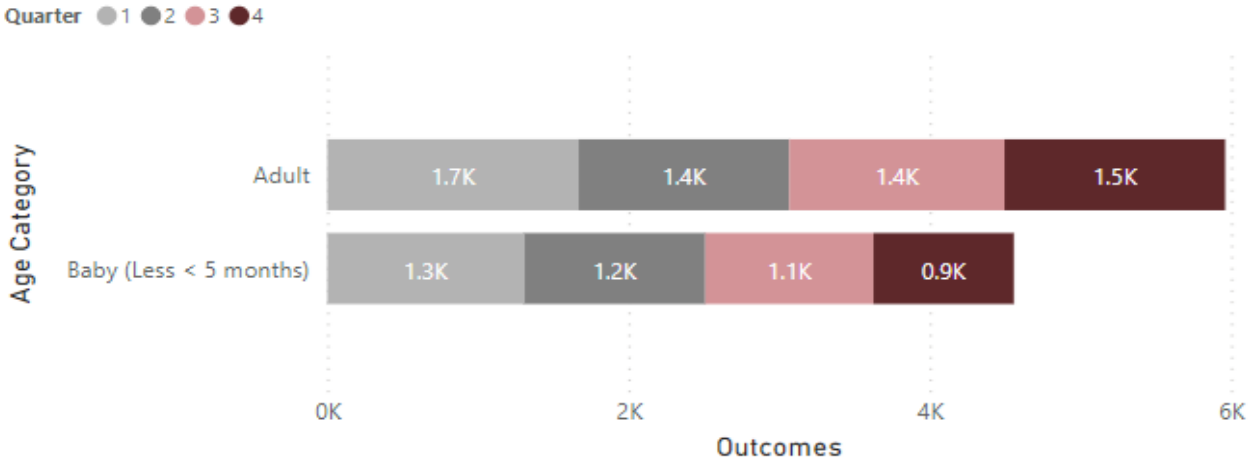
Animal Intake Breakdown by Size Groups



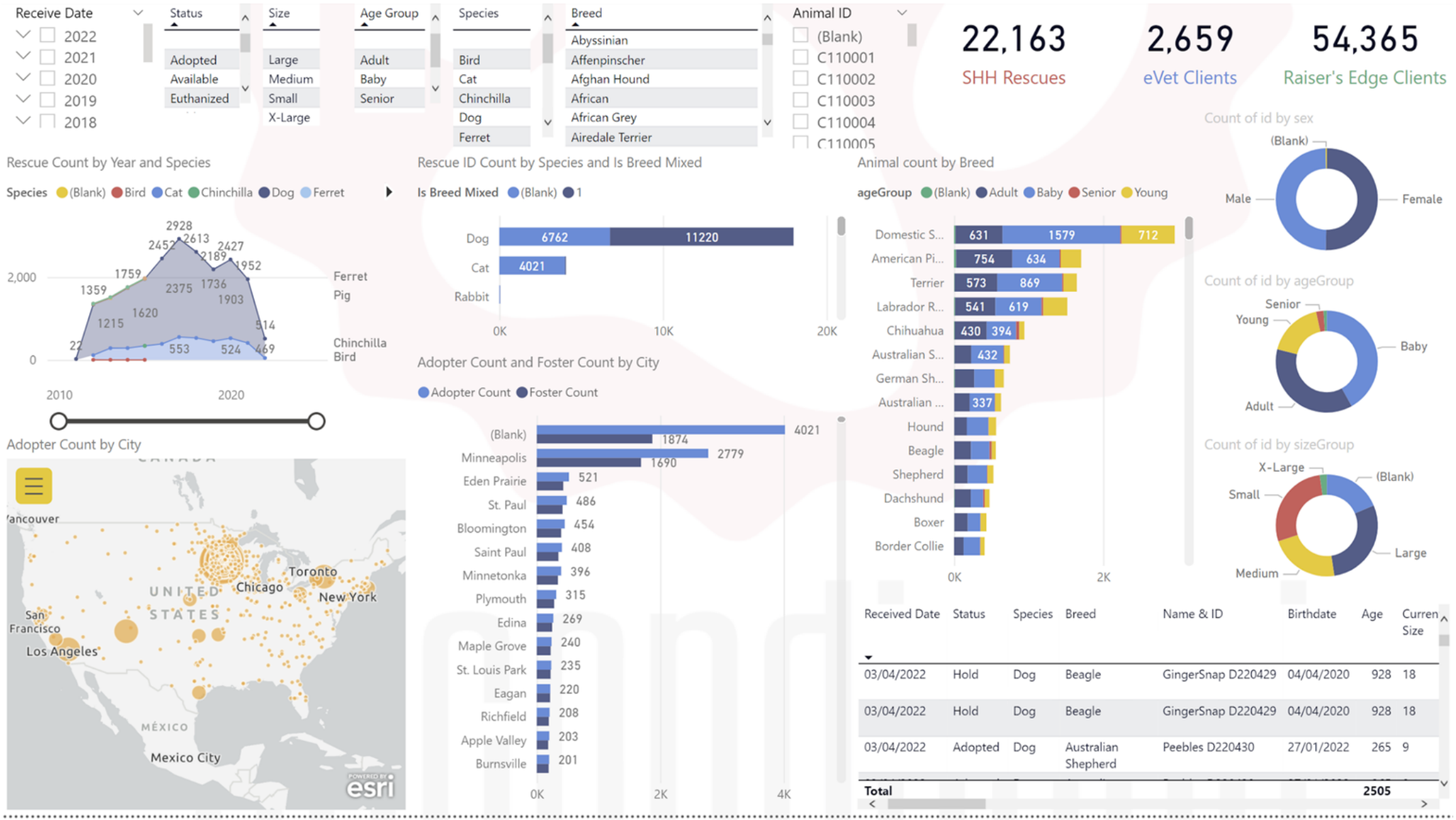
Animal Popularity by Breeds



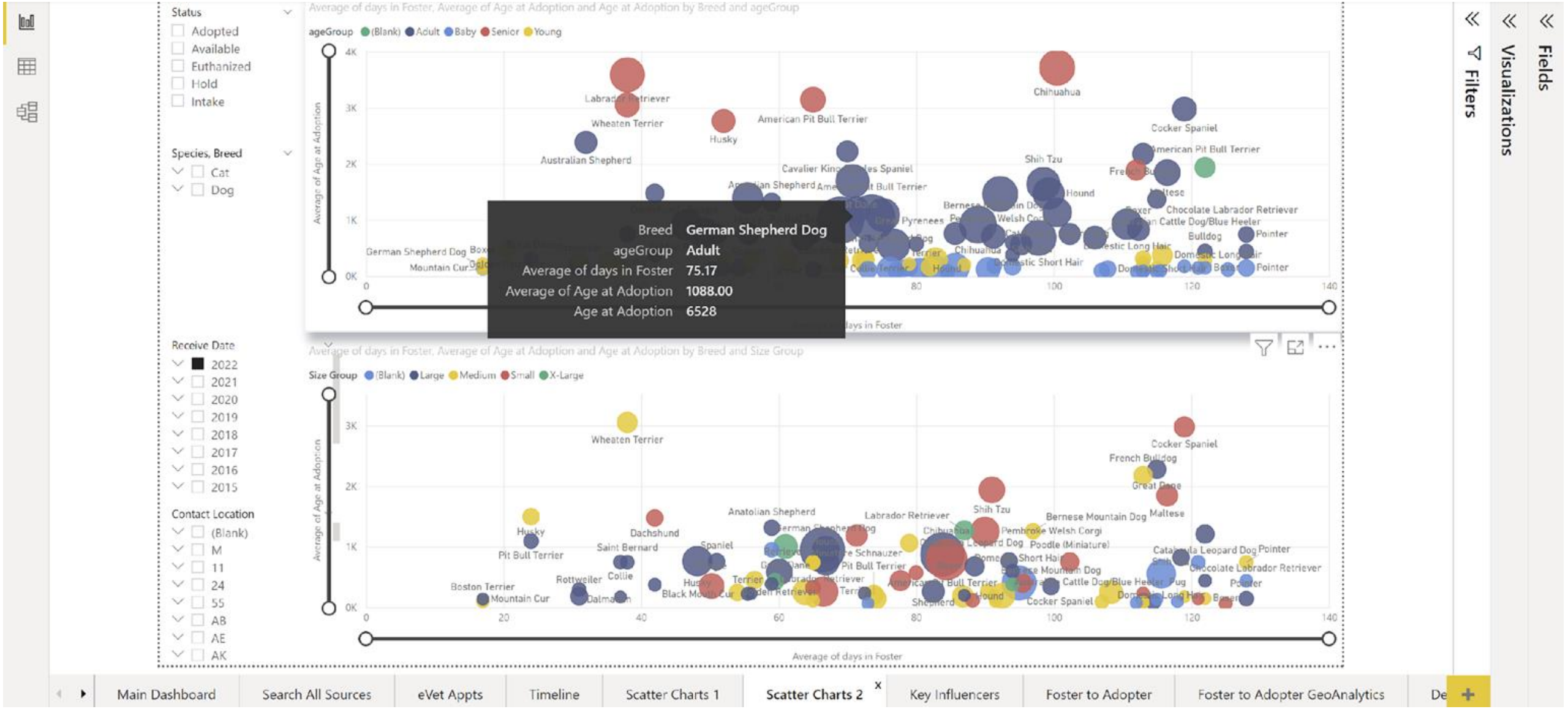
Animal Outcome Trends by Age Category



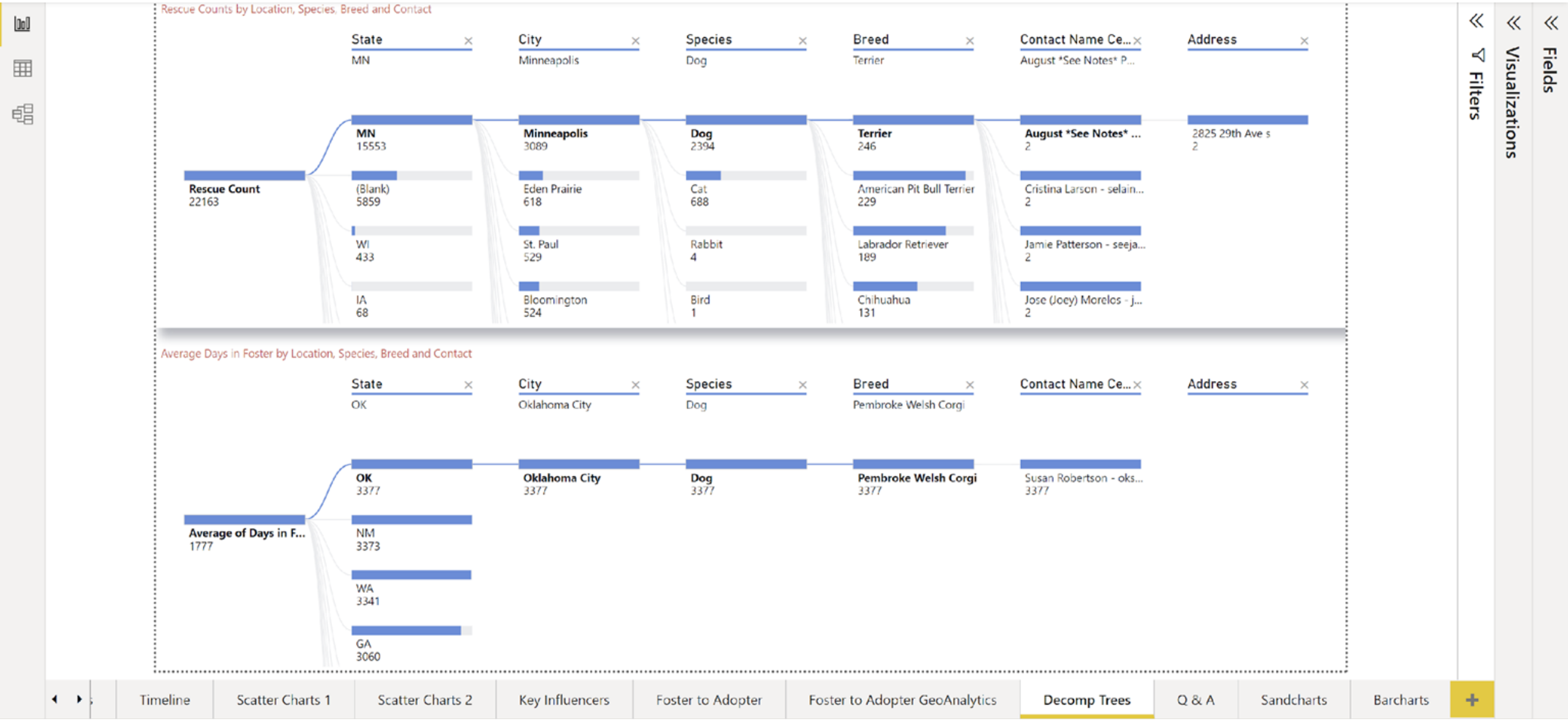
Current Dashboard



Current Dashboard



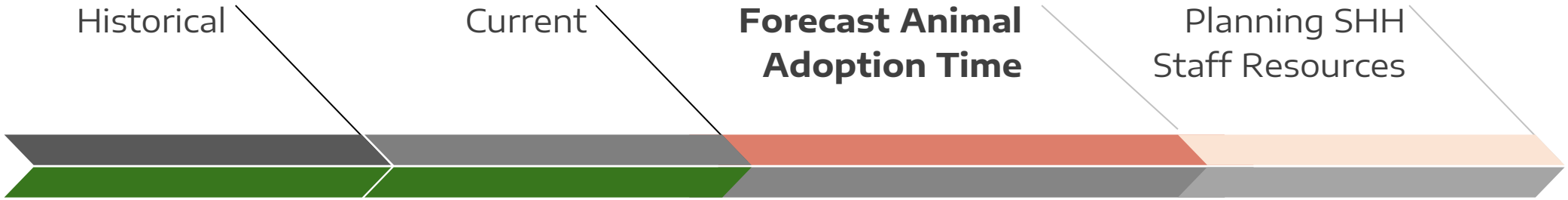
Current Dashboard



Andrei Jula

Python Integration
with MS PowerBI & Google Colab

Animal Rescue as a Nonprofit Organization



Descriptive

Self sufficient, automatic data reloads supporting dashboards in MS PowerBI

What happened?

Historical SHH Animal information, Veterinary appointments, Fosters and Adopters

What is happening?

Current Animal Intakes to be Fostered or Adopted

Predictive

Manually generated models and provided insights

What will happen?

Is the intake to foster or adoption time longer?

Why will it happen?

Are the animal size, location & age at adoption influencing the adoption time?

Prescriptive

Nice to have but not there yet

What should I do?

Plan for additional staff

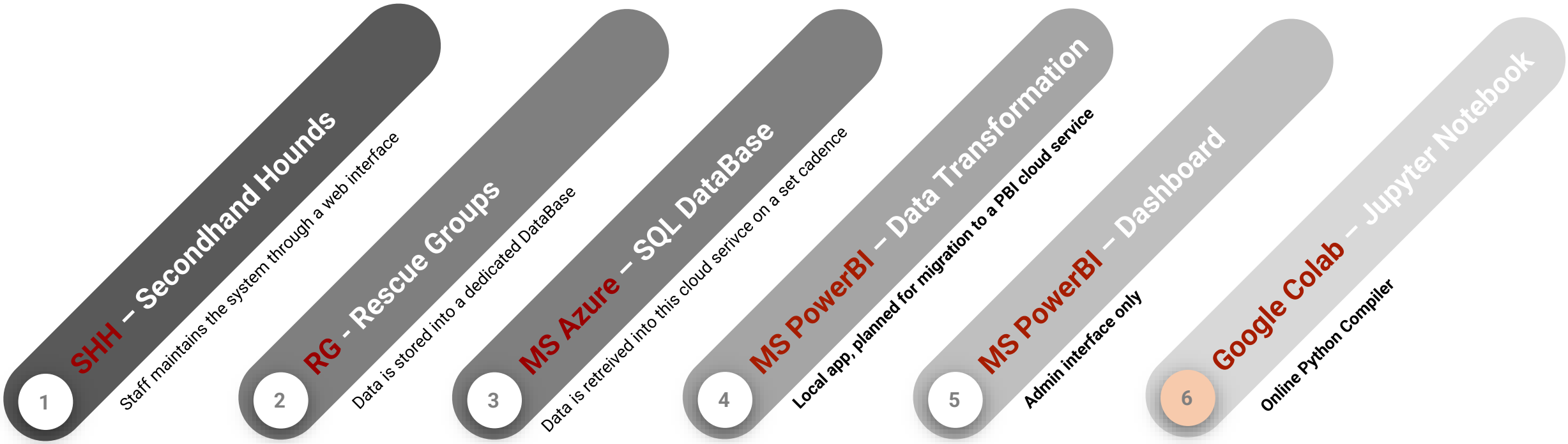
Why should I do it?

Targeted possible adopter reach outs

Secondhand Hound Data Roadmap

Data Entry to Insights

Python Scripts



Automatic data transfer

Python for BI Environments

Pros

- Self sufficient, interactive and user friendly dashboards & reports
- Python visuals can use any Python packages, including custom

Integrations

- MS Visual Studio & Visual Code

Limitations

- Pandas data frames only
- Supported PBI service packages: matplotlib, numpy, pandas, scikit-learn, scipy, seaborn, statsmodels, xgboost
- Maximum 150K rows / 250MB / 5 mins processing
- Charts are not interactive, at 72DPI only
- No column renaming

colab

Pros

- Web service for Jupyter Notebooks
- Main Python libraries already installed and up to date
- Well suited for Machine Learning and data analysis

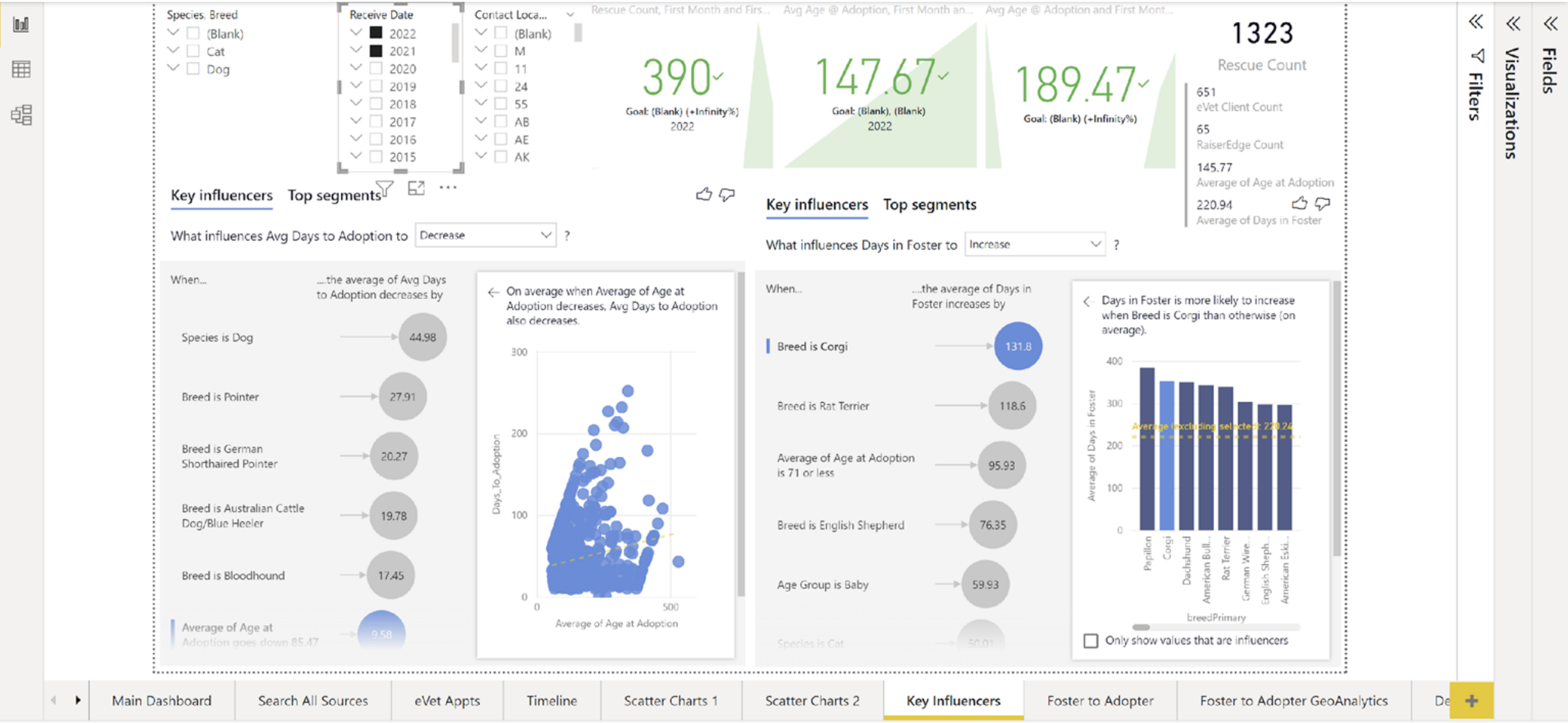
Integrations

- No need for additional code parsers

Limitations

- Not self sufficient for nontechnical business users
- An environment catered for data analysts, requiring custom coding and interpretation

Current Dashboard



Data Preprocessing

Imputation – fill in the missing values, without affecting the model outputs

Correlation Heatmaps – helping identifying which columns cannot be grouped in our tests

Histograms – provide an insight on how data is skewed

Consolidate the date-time information into a single date-time so that we can use it as an index in Pandas

- Define SHH Class Specific Variables
- Impute Missing Data
- Down-sample and Normalize data
- Split Data for Training and Testing

Choosing The Right Data Models

Linear & Polynomial Regression

Regression analysis on relying on dependent and independent variables / predictor and response

Decision Trees using Gini & Entropy – Supervised Machine Learning

Classification Trees

A classification tree is an algorithm where the target variable is categorical, splitting into classes that belong to the response variables such as Yes or No

Regression Trees

A Regression tree is an algorithm where the target variable is continuous.

Simple Neural Network

Flexible, with predictions for binary, numeric and categorical outcomes

Great performance for unstructured data, without the need for linear relationships, handling well outliers

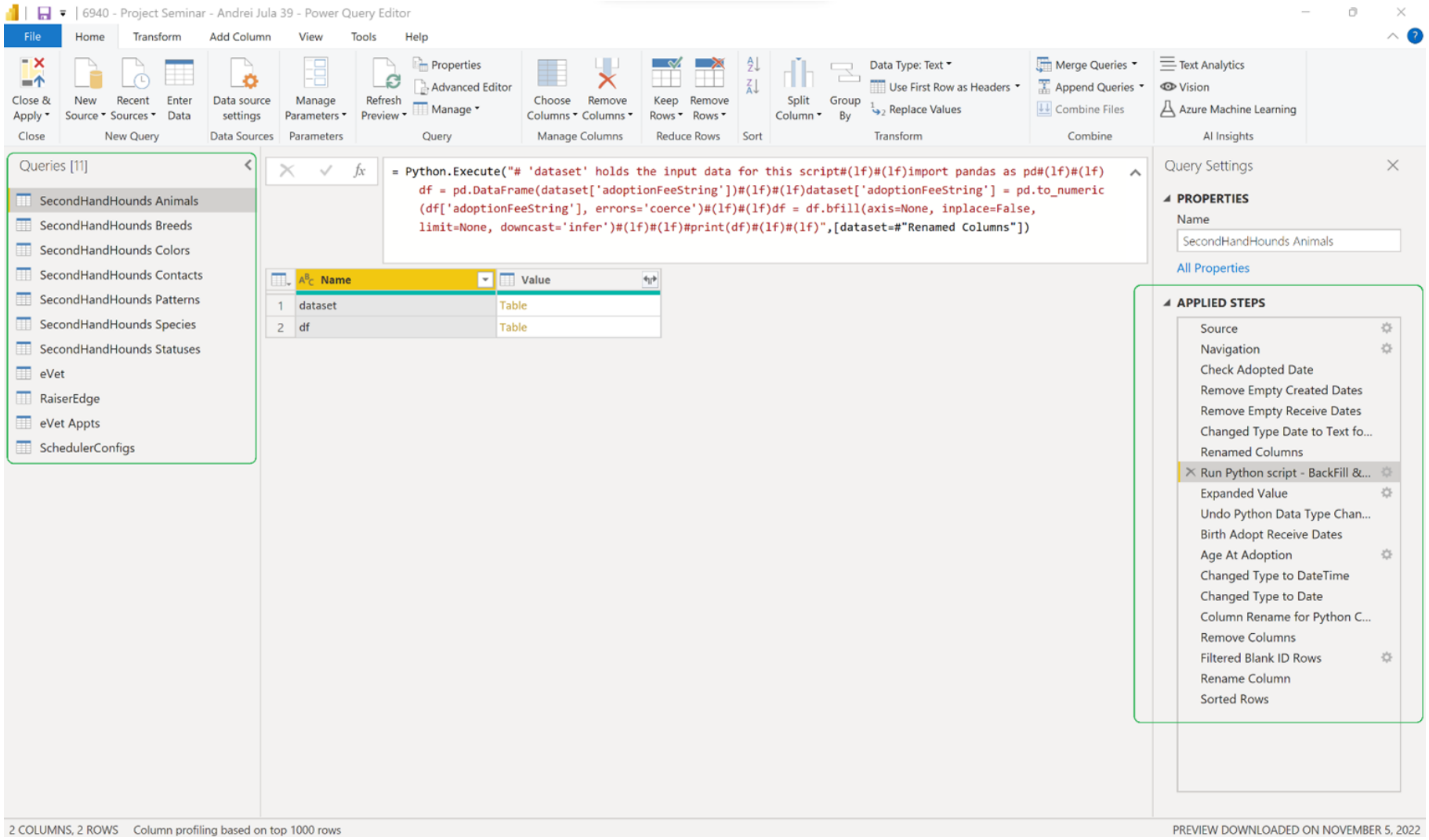
Recurrent Neural Network Long-Short Term Memory Architecture – Deep Learning

Great at Modeling Sequenced / Time Series Data

Effective at pattern learning

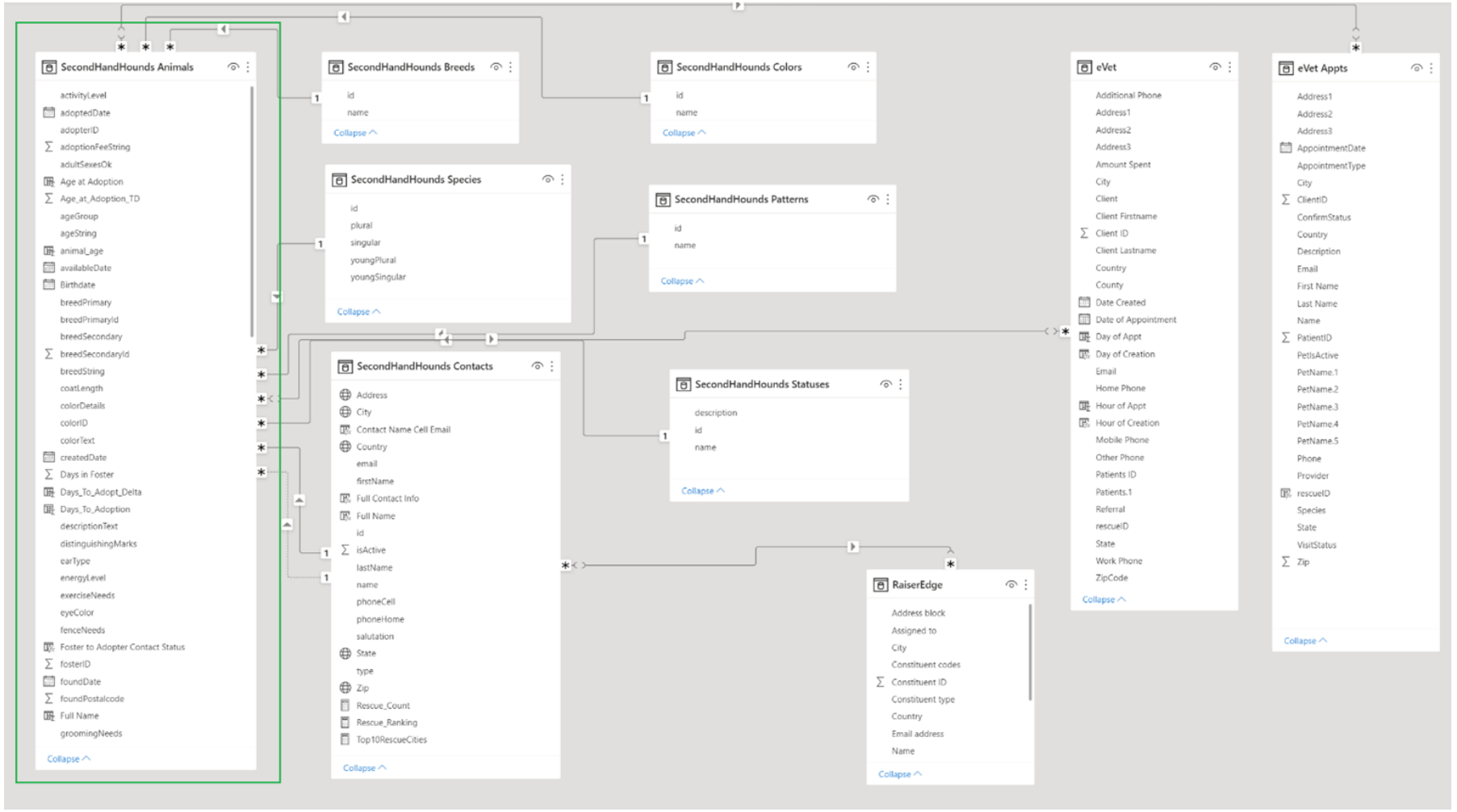
Python Scripts in MS PowerBI

- Python scripts can be inserted in the Transform Data module of MS PowerBI
- The script is ran during the data reload step
- The generated table and custom fields are available for the data model

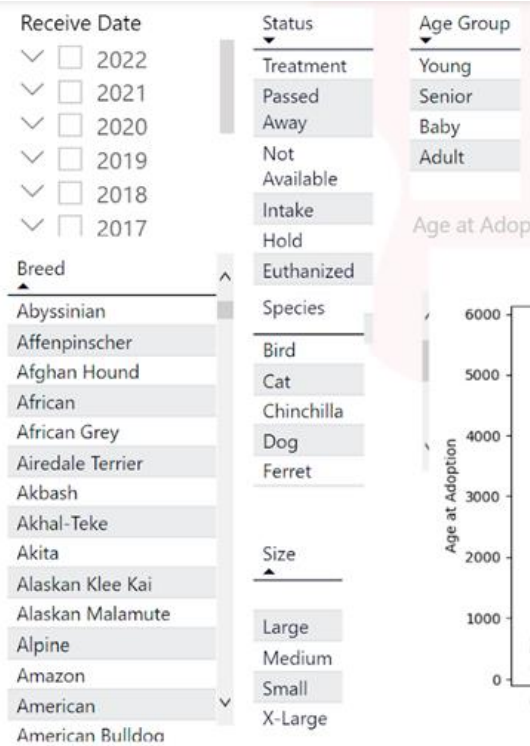


Python Scripts in MS PowerBI

- The generated tables and custom fields are available for the data model



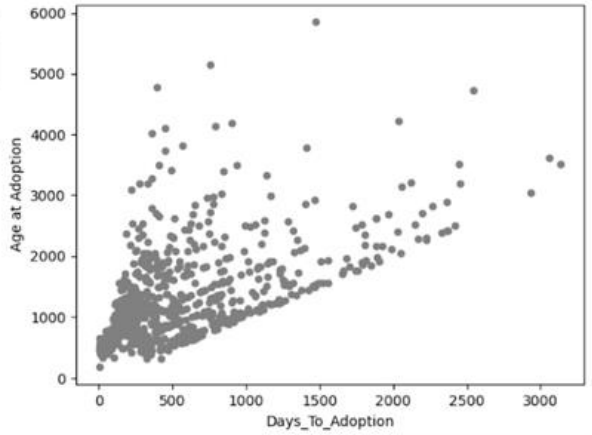
Python Dashboard Objects in MS PowerBI



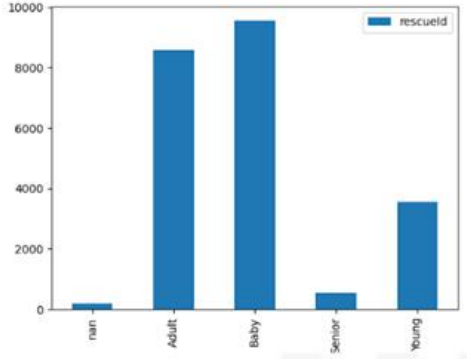
22,432

SHH Rescues

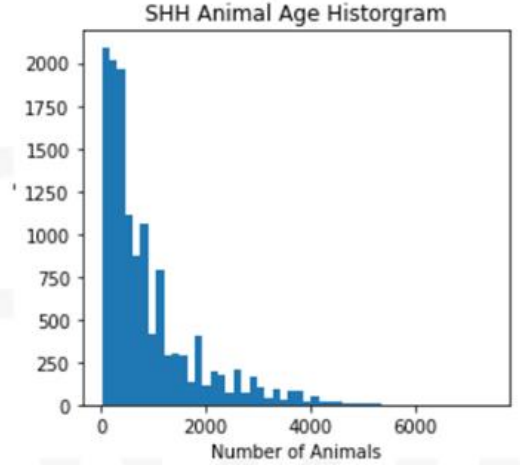
Age at Adoption and Days_To_Adoption



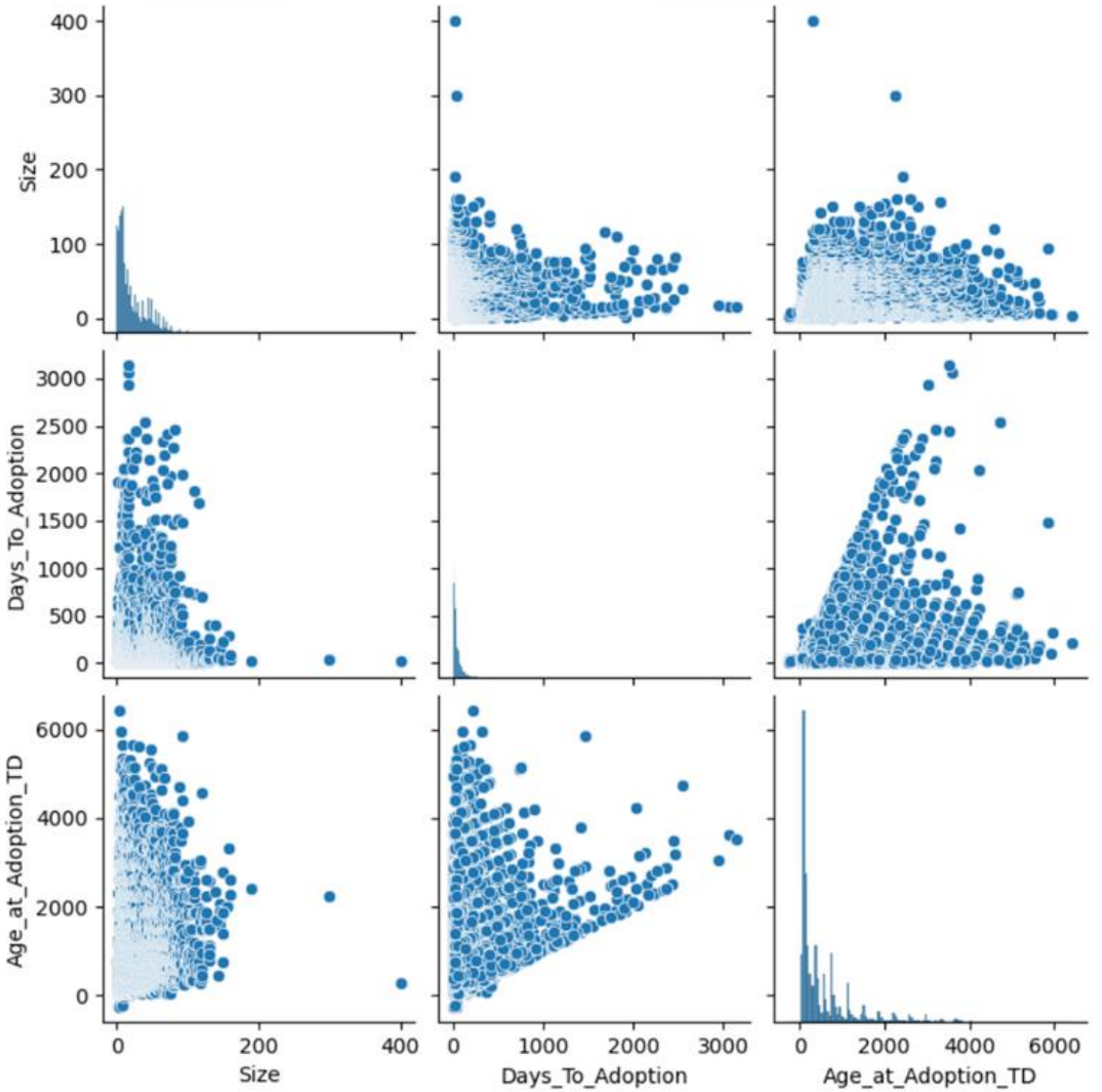
ageGroup and Count of rescueId



Age at Adoption and Size

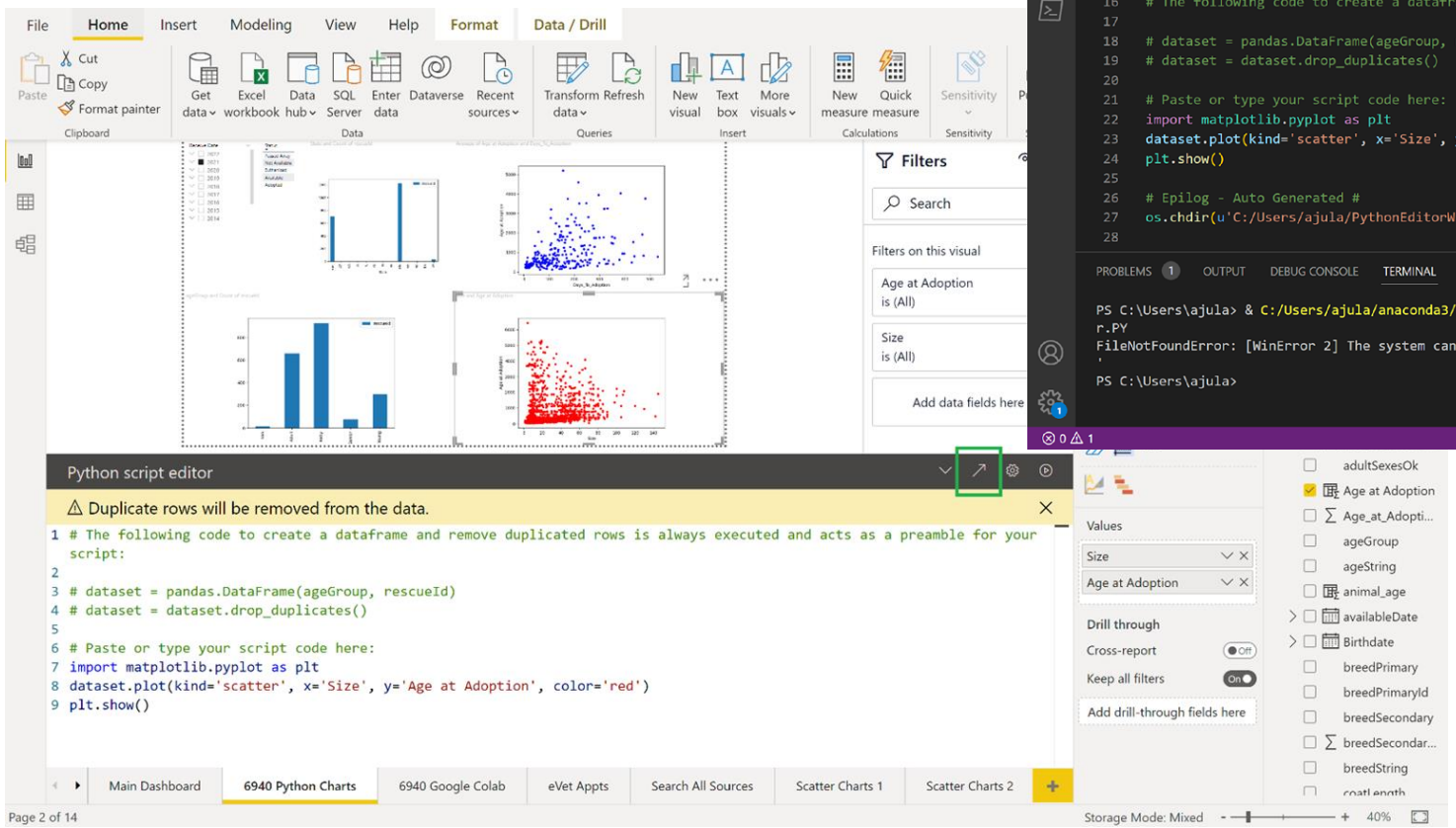


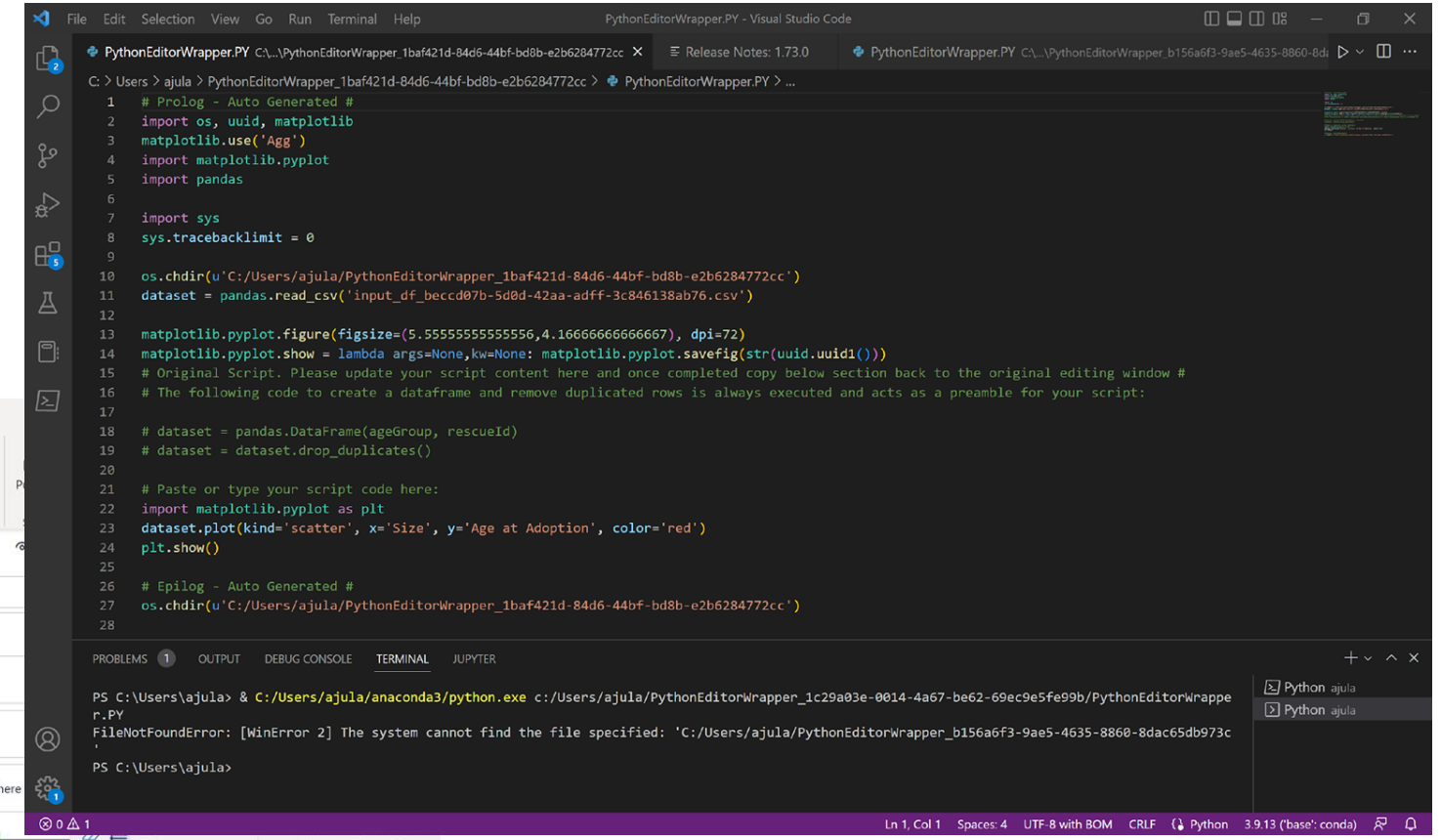
sex, Size, Days_To_Adoption, Age at Adoption, Age_at_Adoption_TD, animal_age, Days in Foster, statusText and Species



MS PowerBI Python Objects – Interpretation in Visual Studio Code

- MS PowerBI can be integrated with MS Visual Studio or MS Visual Code for a friendlier parser environment





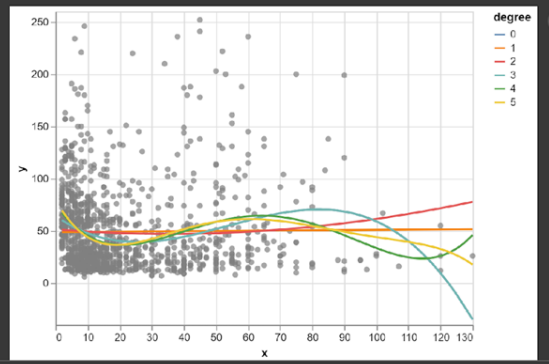
Python Scripts in Google Colab

6940 SHH - Data Prep - 02 Polynomial Regression.ipynb

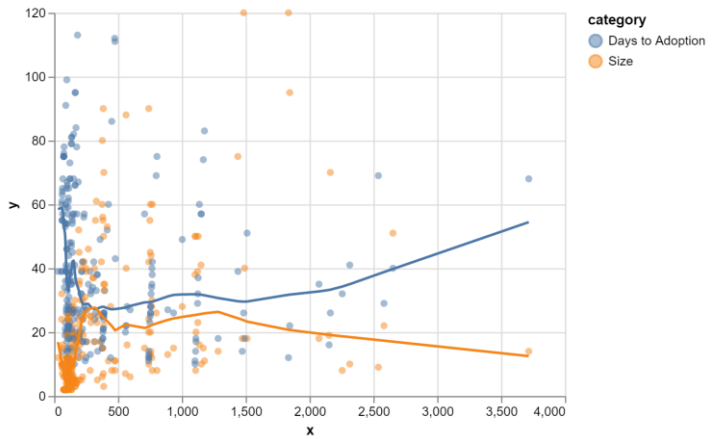
File Edit View Insert Runtime Tools Help

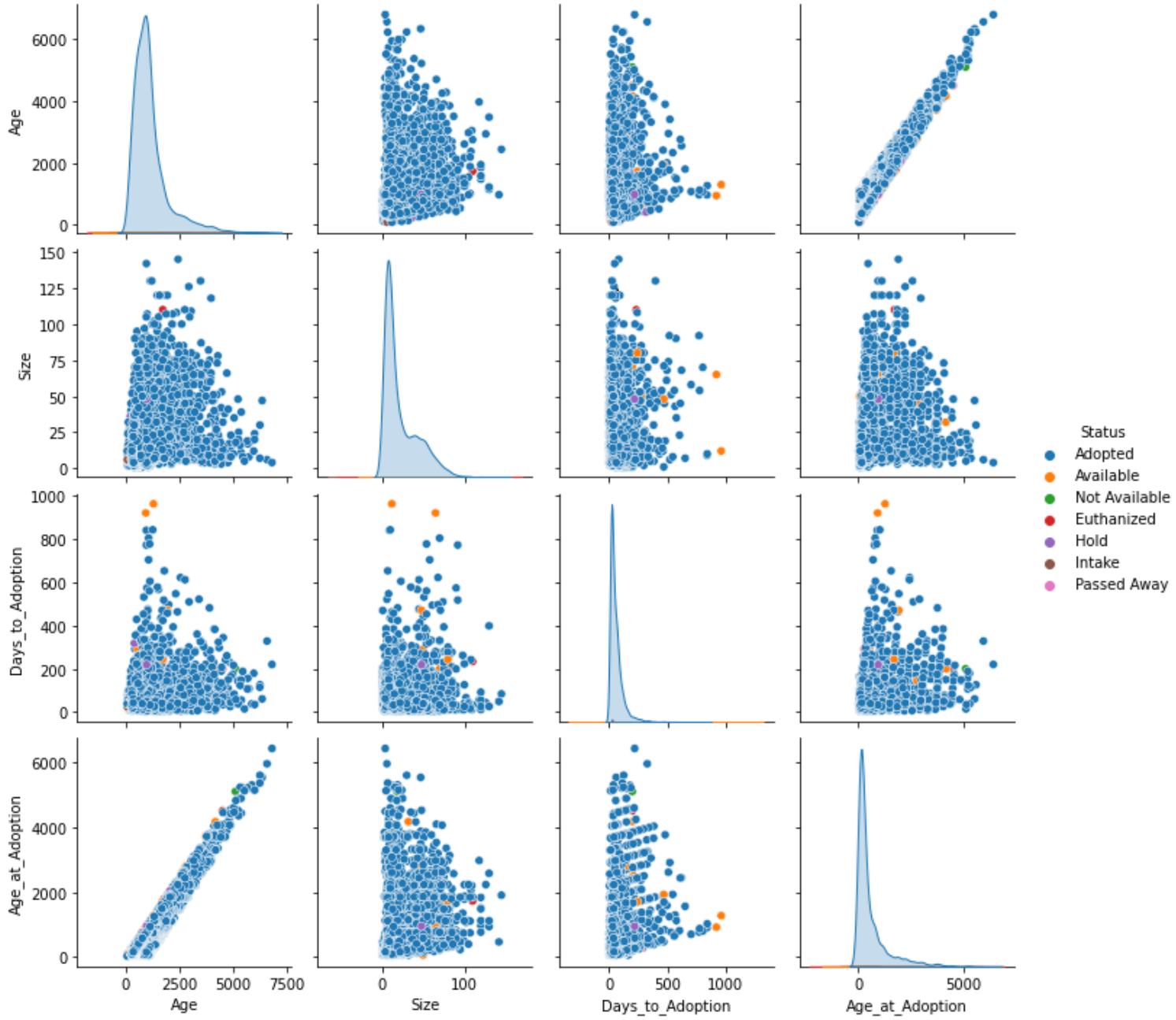
+ Code + Text

```
x = shh.Size
y = shh.Days_to_Adoption
source = pd.DataFrame({"x": x, "y": y})
degree_list = [0, 1, 2, 3, 4, 5] # Define the degree of the polynomial fits
base = alt.Chart(source).mark_circle(color="gray").encode(
        alt.X("x"), alt.Y("y"))
polynomial_fit = [base.transform_regression(
                    "x", "y", method="poly", order=order, as_=["x", str(order)])
                  .mark_line()
                  .transform_fold([str(order)], as_=["degree", "y"])
                  .encode(alt.Color("degree:N"))
                  for order in degree_list]
alt.layer(base, *polynomial_fit)
```



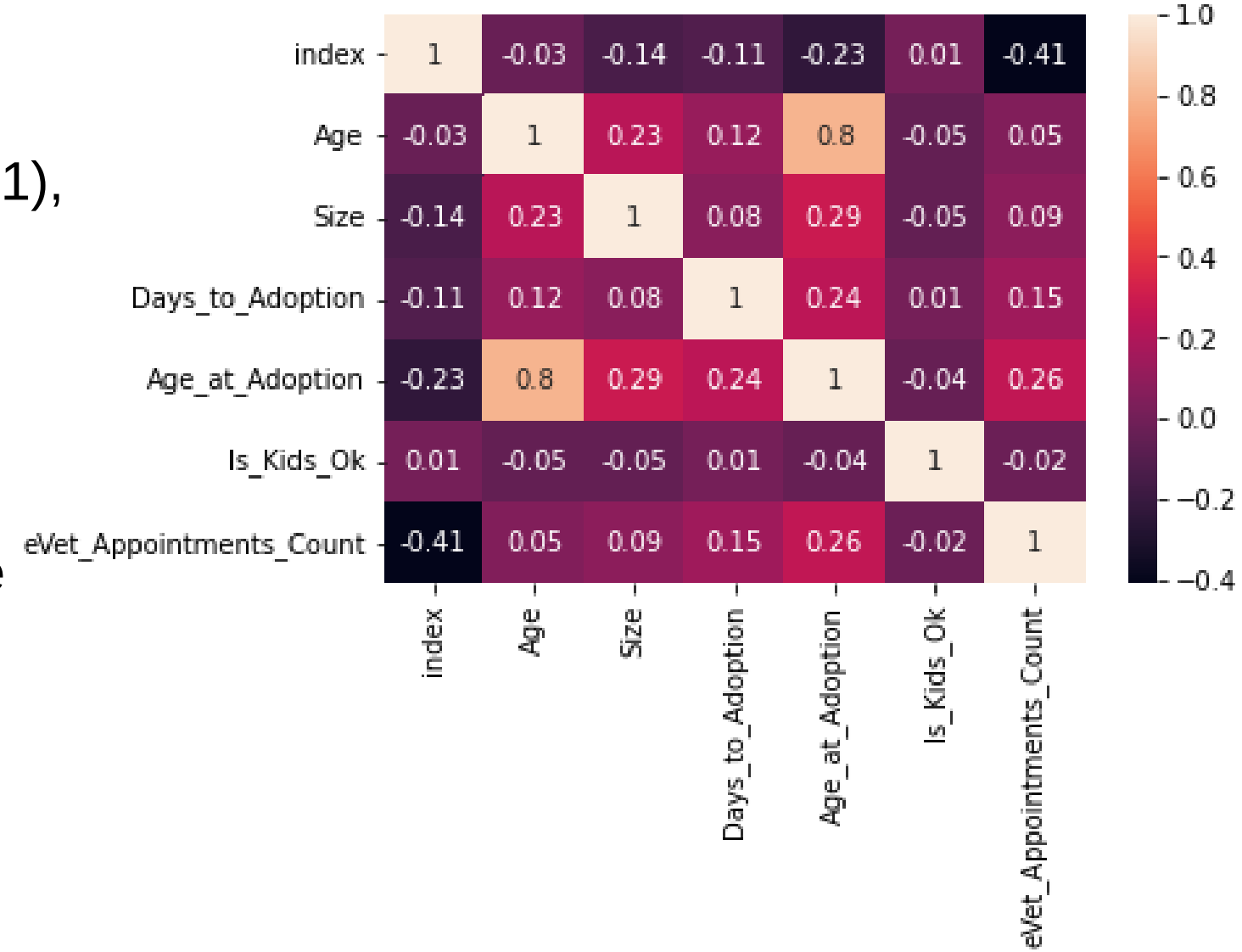
✓ 0s completed at 5:51 PM





SHH Data Heat Map

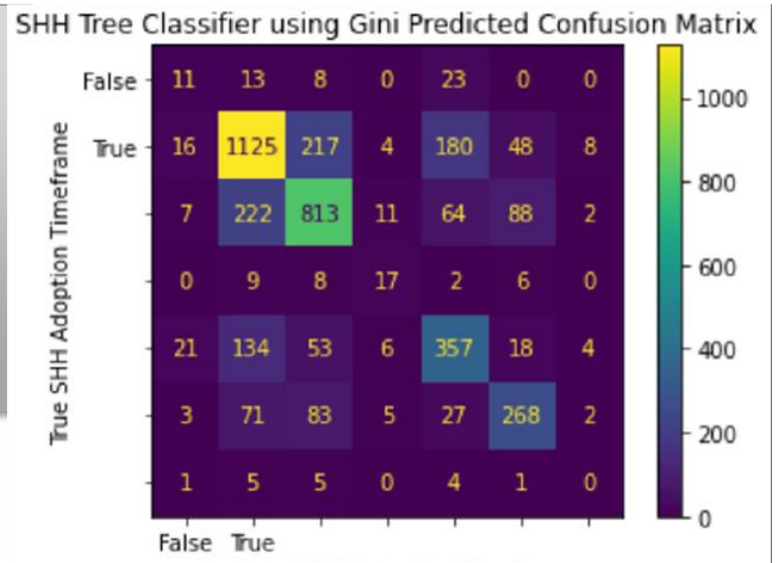
- The closer the value is to 1 (or -1), the stronger a relationship
- The closer a number is to 0, the weaker the relationship
- Negative relationships imply the decrease of a value, while the other increases

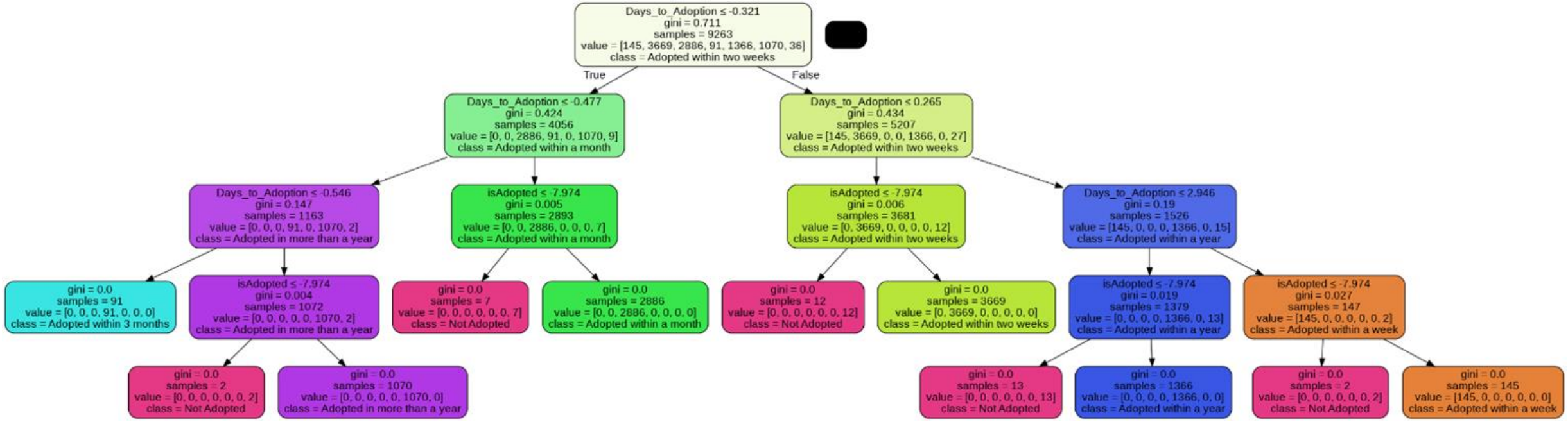


Decision Trees – Supervised Predictive Analytics

Confusion Matrix

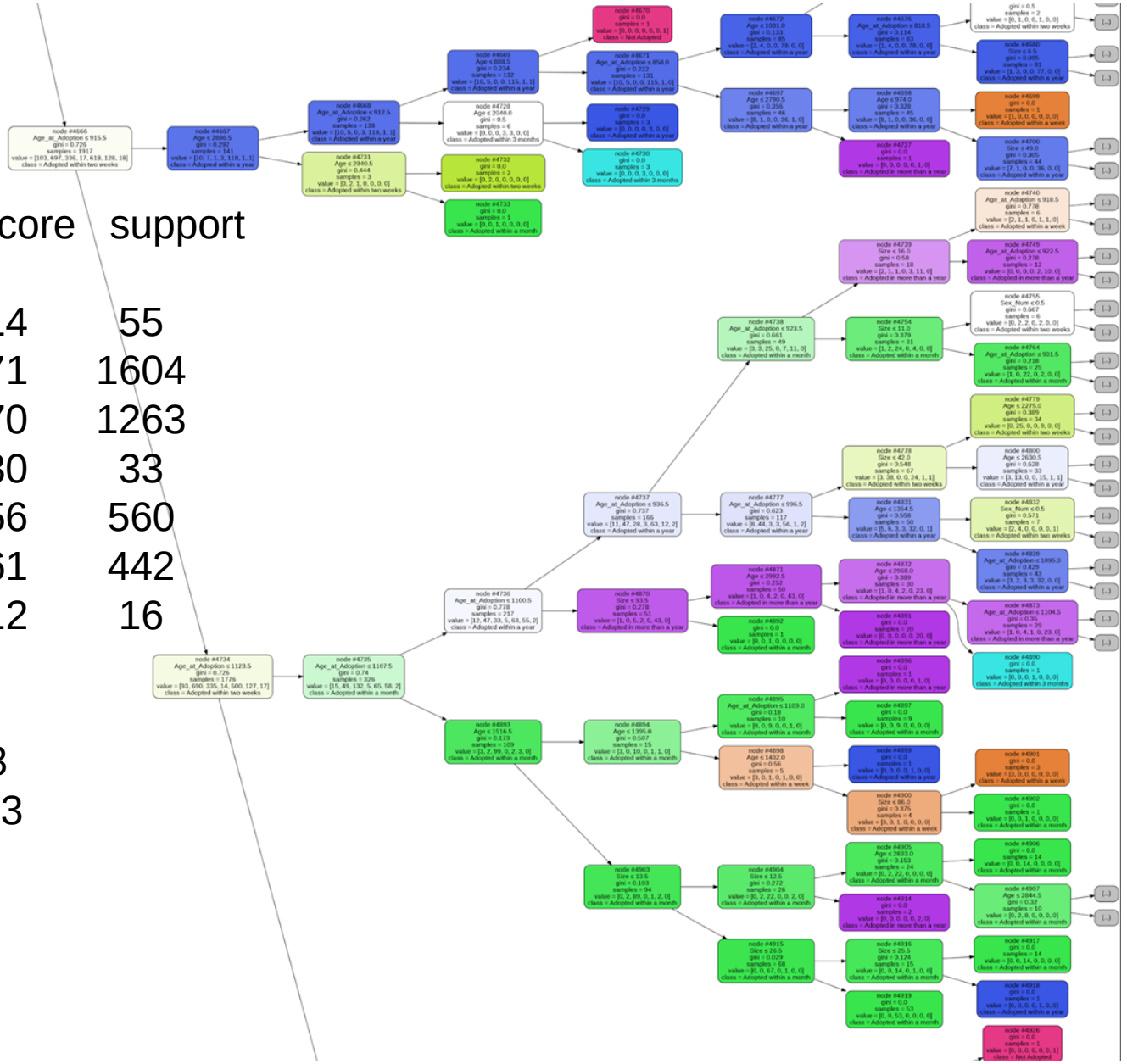
- Accuracy - how often the model is correct: 82.90
- Precision - how well the model made its predictions: 72.16
- Sensitivity / Recall - how well the model is predicting positives: 82.90
- Specificity - how well the model is predicting negatives: 82.90
- F-Score - harmonic mean of precision and sensitivity: 75.29





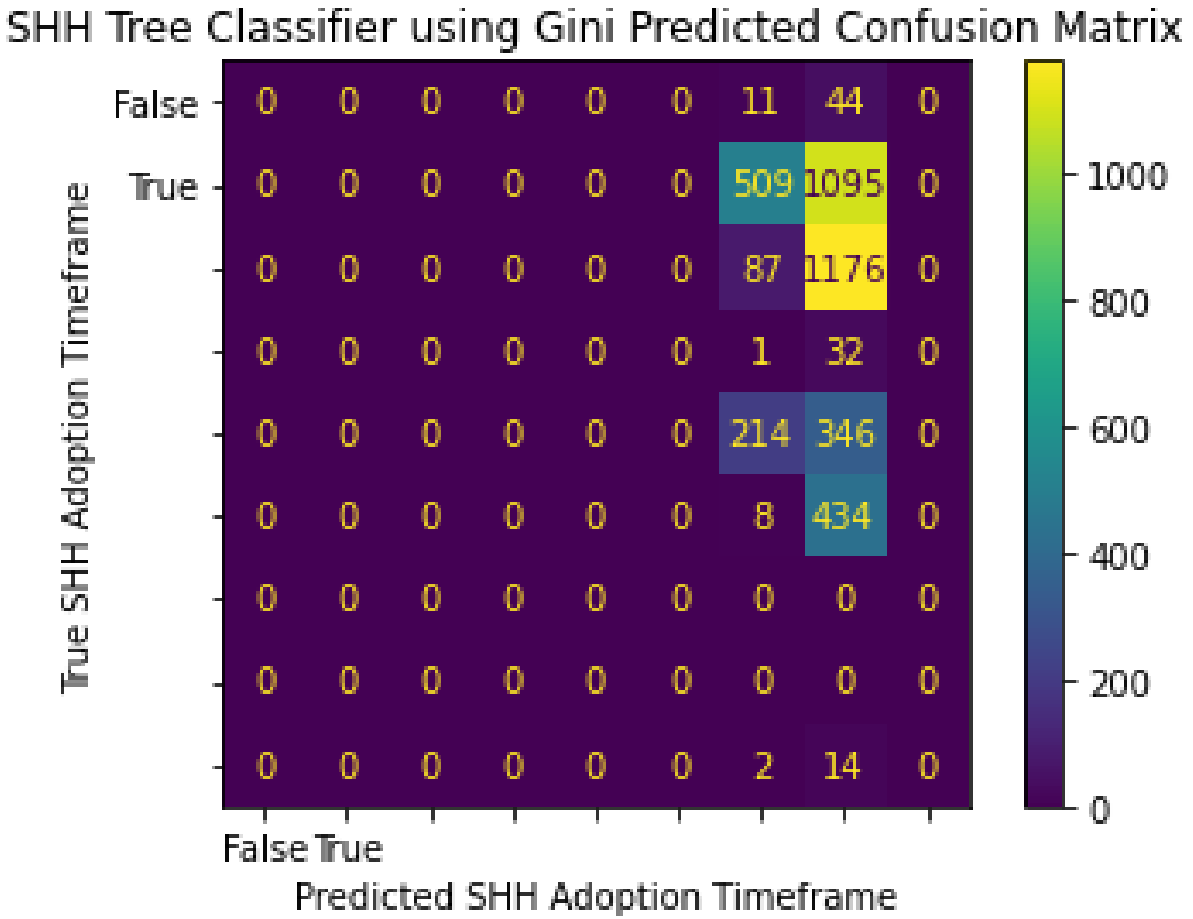
Python Scripts in MS PowerBI

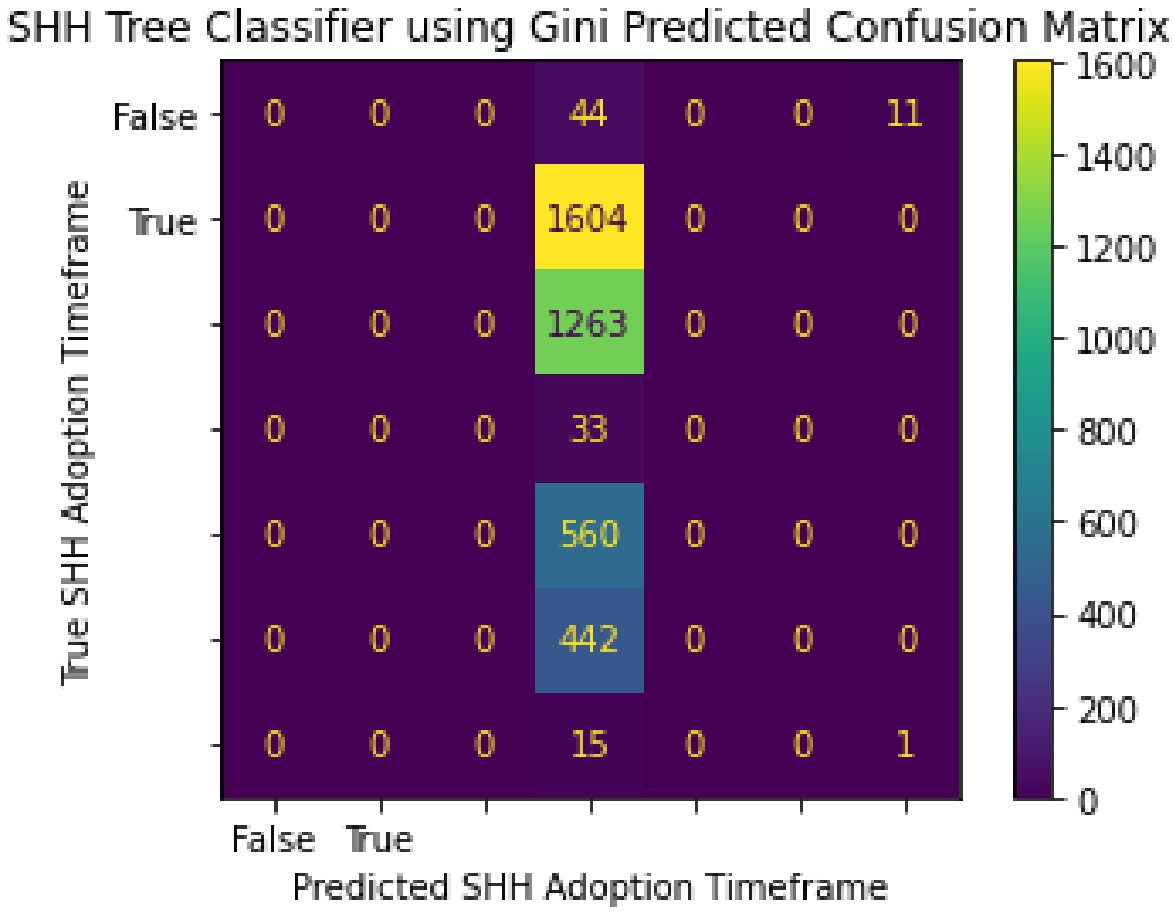
Report :	precision	recall	f1-score	support
Adopted in more than a year	0.13	0.15	0.14	55
Adopted within 3 months	0.72	0.70	0.71	1604
Adopted within a month	0.71	0.70	0.70	1263
Adopted within a week	0.26	0.36	0.30	33
Adopted within a year	0.53	0.59	0.56	560
Adopted within two weeks	0.61	0.60	0.61	442
Not Adopted	0.11	0.12	0.12	16
accuracy			0.66	3973
macro avg	0.44	0.46	0.45	3973
weighted avg	0.66	0.66	0.66	3973



Confusion Matrix from a saved Model

Confusion Matrix from a saved Model automatically dropped null rows, repositioned the true vs false matrix and added the values within the same row, without affecting the ability to predict





ML Simple Neural Network

6940 SHH - 05 - ML Simple Neural Network.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

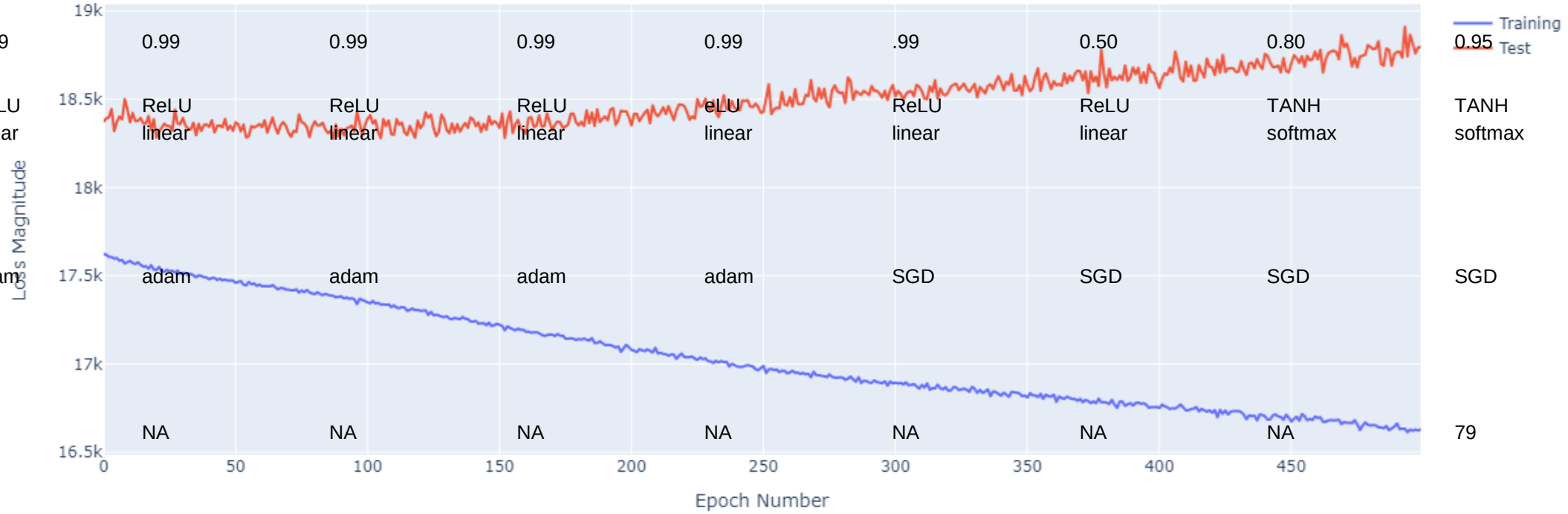
Connect

False

```
print('Defining compiling parameters for tweaking the model to its best (smallest) loss')
model = Sequential()

# Model compiling parameters

activation_hidden = 'relu' #'elu' 'tanh'
activation_output = 'linear' #'softmax'

print('Defining the hidden dense layers, one or many, with different number of neurons, using different activation functions')
model.add (Dense(32, input_shape=(6,), activation=activation_hidden, name='dense_1'))
model.add (Dense(16, activation=activation_hidden, name='dense_2'))
#model.add (Dense(32, activation=activation_hidden, name='dense_3'))

print('Defining the output dense layer as a linear or softmax activation')
model.add (Dense(1, activation = activation_output, name = 'output'))

print('We are using the adam (Adaptive Moment Estimation) optimizer as baseline')
#optimizer='adam' #'rmsprop'
optimizer = tf.keras.optimizers.Adam(lr = 0.0001)

# Learning Rate and Momentum variables used in the SGD optmizer only
lr = 0.0001
momentum = 0.99
#optimizer = SGD(lr=lr, momentum=momentum)

print('In order to know if the model is properly learning, we use a mean squared error (‘mse’) loss function')
loss='mse'
#loss='categorical_crossentropy'

print('To report the performance of it we will adopt the mean average error (‘mae’) metric')
#metrics='accuracy'
metrics='mae'
#metrics='mse'
#metrics='mape'

model.compile(optimizer, loss, metrics)

print('Using the summary method from Keras, we are printing the model structure')
model.summary()
```

Simple Neural Network

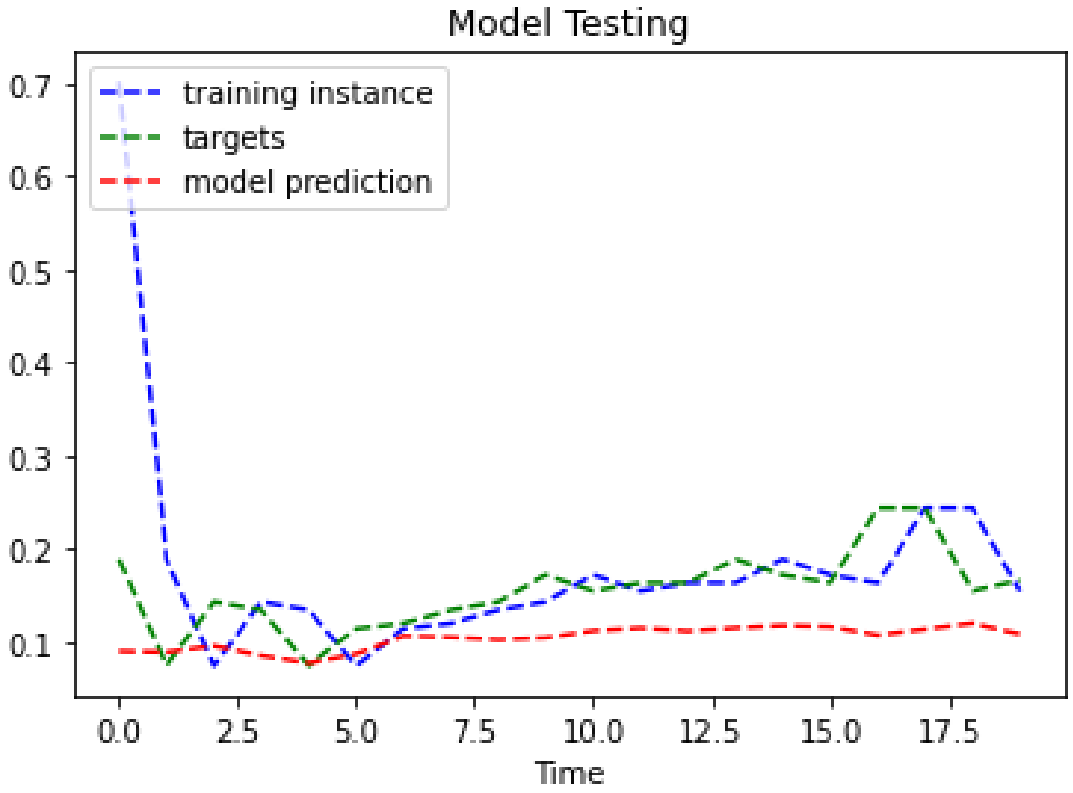
Number of neurons in the hidden layer(s)	Layer 1 = 16	Layer 1 = 16 Layer 2 = 16	Layer 1 = 32 Layer 2 = 16 Layer 3 = 16	Layer 1 = 16 Layer 2 = 16 Layer 3 = 32	Layer 1 = 128 Layer 2 = 64 Layer 3 = 32	Layer 1 = 64 Layer 2 = 32	Layer 1 = 32 Layer 2 = 16 Layer 3 = 32	Layer 1 = 128 Layer 2 = 64 Layer 3 = 32	Layer 1 = 128 Layer 2 = 64 Layer 3 = 32
Learning Rate	0.001	0.001	0.001	0.00001	0.001	0.0001	0.001	0.001	0.001
Momentum	0.99	0.99	0.99	0.99	0.99	.99	0.50	0.80	0.95
Activation Functions Hidden / Output	ReLU linear	ReLU linear	ReLU linear	ReLU linear	eLU linear	ReLU linear	ReLU linear	TANH softmax	TANH softmax
Optimizers Adaptive Moment Estimation / Stochastic Gradient Descent	adam	adam	adam	adam	adam	SGD	SGD	SGD	SGD
Early Stop Epoch Count Patience as 10	NA	NA	NA	NA	NA	NA	NA	NA	79
Time in seconds	125	134	114	143	132	143	145	145	223
Loss as MSE/MAE	loss: 17885.29 mae: 55.53	loss: 17924.82 mae: 55.88	loss: 17941.82 mae: 55.74	loss: 18515.61 mae: 58.81	loss: 18031.17 mae: 56.97	loss: 20105.48 mae: 58.62	loss: 18562.87 mae: 57.24	loss: 0.2194 mae: 56.17	loss: 24755.57 mae: 68.20



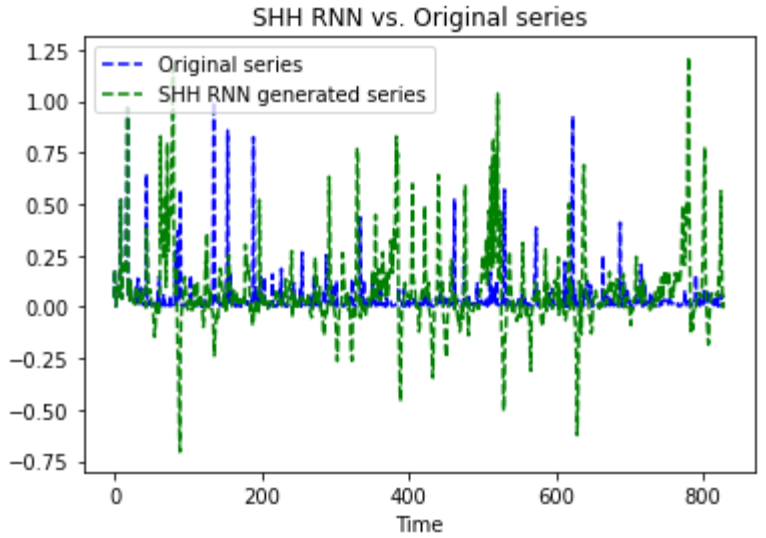
Recurring Neural Network Long-Short Term Memory Architecture

RNN LSTM Model using *ADAM (Adaptive Moment Estimation) as an optimizer and Mean Squared Error as loss suppression

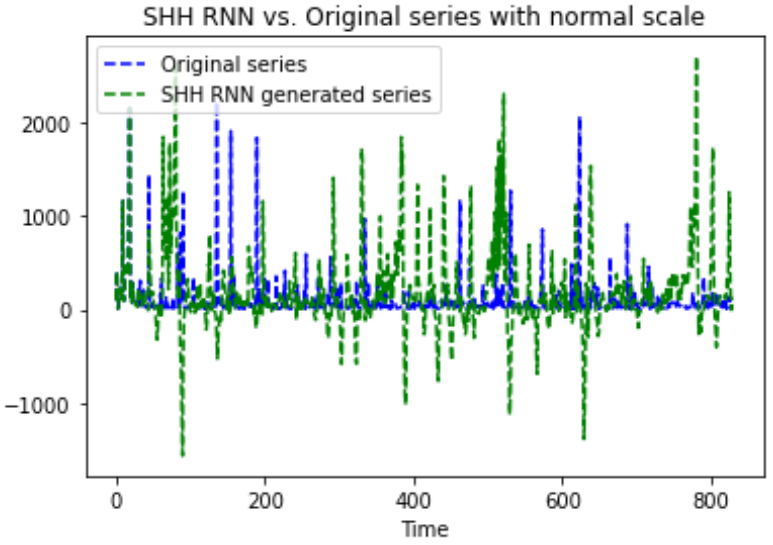
ADAM - Use of an exponentially decaying average of past gradients



Model Plot on Predicted Sequence on SHH Data since 01/01/2020



Plot on normalized Days to Adoption Scale



Plot on inversed to normal Days to Adoption Scale

Resources / Bibliography

2023 Great Lakes Data & Analytics Summit - Call for Speakers (witinc.com)

Organization Info

1. SHH Organization retrieved from Secondhand Hounds | Animal Rescue | Twin Cities, Minnesota
2. SHH Contacts retrieved from Our Team | Secondhand Hounds
3. Oakland University News, School of Business Administration retrieved from Data For Good: Project delivers outcomes to benefit animal rescue efforts (oakland.edu)

Environments

1. Setting the Python/Pandas/MS Visual Studio Environment steps retrieved from Run Python Scripts in Power BI Desktop - Power BI | Microsoft Learn Environment
2. Run Python Scripts in Power BI Desktop - Power BI | Microsoft Learn
3. Create Power BI visuals using Python in Power BI Desktop - Power BI | Microsoft Learn
4. Installation - pip documentation v22.3 (pypa.io)
5. Python in Visual Studio Code as an Extension in Visual Studio Code – Optional, but highly recommended for an intuitive parser, automatic code backup with Microsoft or GitHub, Intelligent Code Completion, variables, methods and imported modules, debugging, source control, etc.
6. Documentation for Visual Studio Code
7. Use an external Python IDE with Power BI - Power BI | Microsoft Learn
8. The Python Tutorial — Python 3.10.8 documentation
9. Pillow · PyPI
10. How To Fix The Python Pillow Error ImportError: DLL load failed while importing _imaging: The specified module could not be found. (dev2qa.com)

Data Preparation

1. What is One Hot Encoding and How to Do It | by Michael DelSole | Medium

Decision Trees

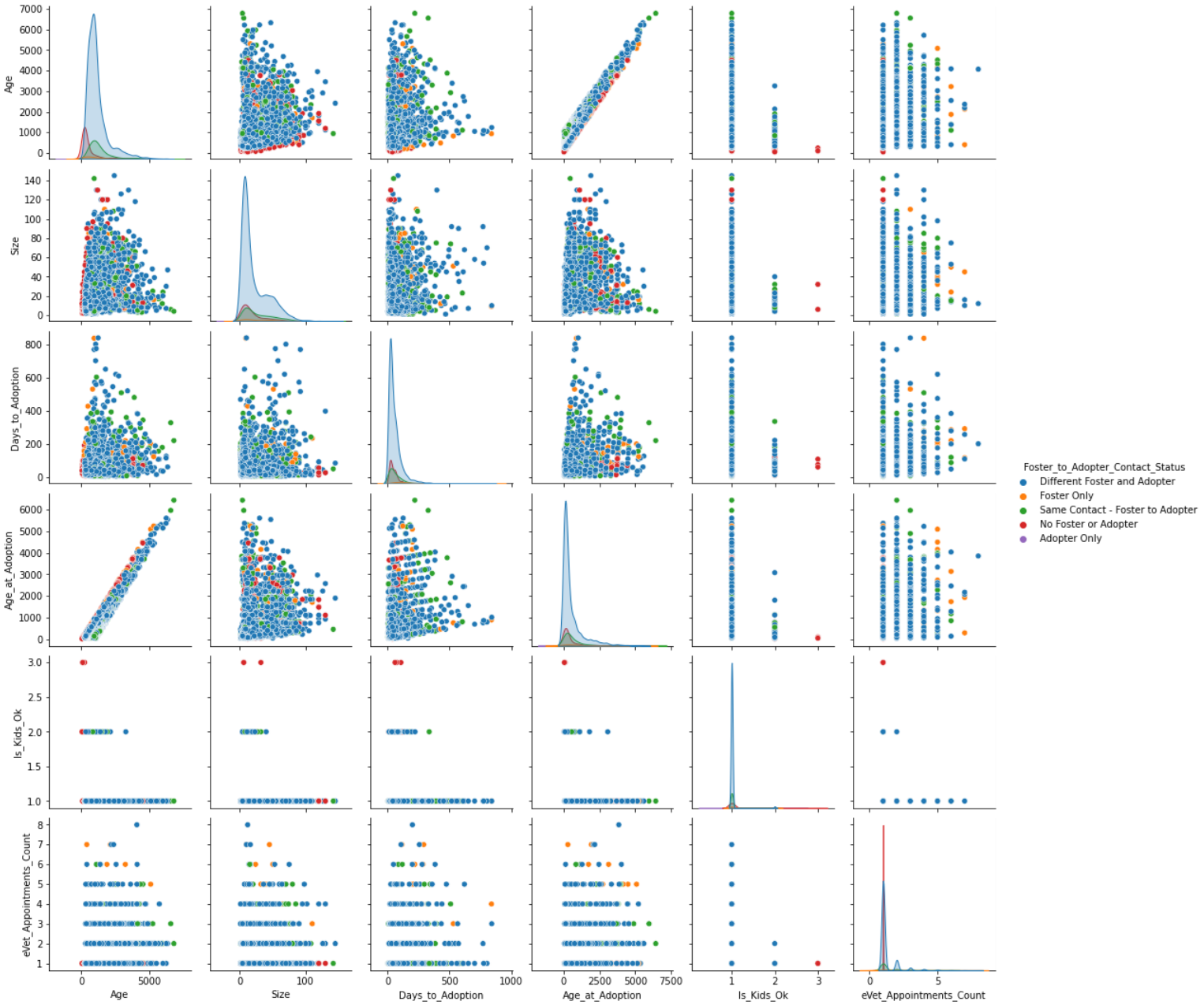
1. Python Decision Tree Classification Tutorial: Scikit-Learn DecisionTreeClassifier | DataCamp
2. Decision Tree Classifier - The Click Reader
3. Decision Tree Classifier with Sklearn in Python • datagy
4. Decision Trees: Gini vs Entropy | Quantdare
5. python - Facing ValueError: Target is multiclass but average='binary' - Stack Overflow
6. CART (Classification And Regression Tree) in Machine Learning – GeeksforGeeks
7. Random Forest in Python. A Practical End-to-End Machine Learning… | by Will Koehrsen | Towards Data Science
8. Decision Tree Classifier explained in real-life: picking a vacation destination | by Carolina Bento | Towards Data Science

Machine Learning

1. Python Machine Learning - Confusion Matrix (w3schools.com)
2. Python Machine Learning Normal Data Distribution (w3schools.com)
3. Python Machine Learning Decision Tree (w3schools.com) -> W3Schools Tryit Editor
4. Time Series Prediction with LSTM Recurrent Neural Networks in Python with Keras - MachineLearningMastery.com

Backup

Data Correlation – Pairplots



Data Correlation – Pairplots

- Animal Size
- Age
- Days to Adoption
- Age at Adoption
- Is Kid Friendly
- Number of eVet Appointments

By highlighting the different statuses:

- Different Foster and Adopter
- Foster Only
- Same contact - Foster to Adopter
- No Foster or Adopter
- Adopter Only

Python in Google Colab, new logical columns

Create New SHH Logical Columns

```
# Create a new column True / False that represents the animal adoption status as binary
# Create additional SHH adoption segments in logical increments

shh['Is_Kids_Ok_TF'] = np.where(shh['Is_Kids_Ok'] == 1, True, False)

shh['isAdopted'] = np.where(shh['Status'] == 'Adopted', True, False)

shh['isAdopted_7'] = np.where((shh['Status'] == 'Adopted') & (shh['Days_to_Adoption'] <= 7), True, False)
shh['isAdopted_14'] = np.where((shh['Status'] == 'Adopted') & (shh['Days_to_Adoption'] > 7) & (shh['Days_to_Adoption'] <= 14), True, False)
shh['isAdopted_30'] = np.where((shh['Status'] == 'Adopted') & (shh['Days_to_Adoption'] > 14) & (shh['Days_to_Adoption'] <= 30), True, False)
shh['isAdopted_90'] = np.where((shh['Status'] == 'Adopted') & (shh['Days_to_Adoption'] > 30) & (shh['Days_to_Adoption'] <= 90), True, False)
shh['isAdopted_365'] = np.where((shh['Status'] == 'Adopted') & (shh['Days_to_Adoption'] > 90) & (shh['Days_to_Adoption'] <= 365), True, False)

shh['isAdopted_7_365'] = np.where((shh['Status'] == 'Adopted') & (shh['Days_to_Adoption'] <= 7), 'Adopted within a week',
                         np.where((shh['Status'] == 'Adopted') & (shh['Days_to_Adoption'] > 7) & (shh['Days_to_Adoption'] <= 14), 'Adopted within two wee
                         np.where((shh['Status'] == 'Adopted') & (shh['Days_to_Adoption'] > 14) & (shh['Days_to_Adoption'] <= 30), 'Adopted within a mont
                         np.where((shh['Status'] == 'Adopted') & (shh['Days_to_Adoption'] > 30) & (shh['Days_to_Adoption'] <= 90), 'Adopted within 3 mont
                         np.where((shh['Status'] == 'Adopted') & (shh['Days_to_Adoption'] > 90) & (shh['Days_to_Adoption'] <= 365), 'Adopted within a yea
                         np.where((shh['Status'] == 'Adopted') & (shh['Days_to_Adoption'] > 365), 'Adopted in more than a year',
                                  'Not Adopted'))))) # catch all timeframes

```

Data Prep – Understanding SHH Data, Animal Age Histogram

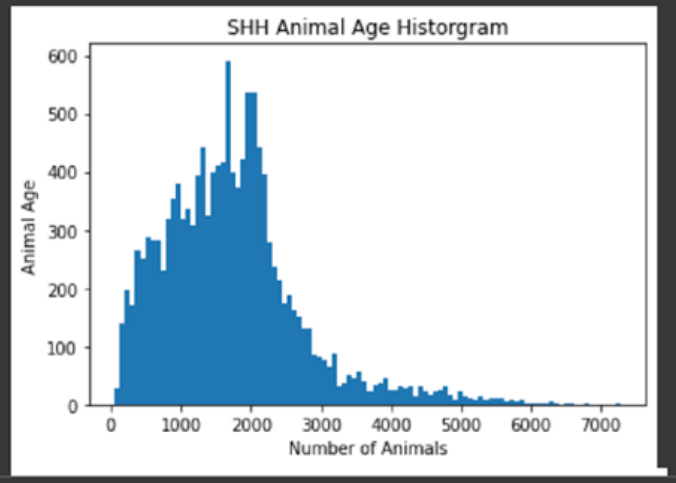
6940 SHH - Data Prep - 00 Understanding SHH DataSet.ipynb

File Edit View Insert Runtime Tools Help Last edited on November 24

Comment Share

+ Code + Text Connect Editing

```
# Second Hand Hound Animal Age at Adoption histogram with 100 bars
plt.hist(shh.Age_at_Adoption, 100)
plt.title("SHH Animal Age at Adoption Historgram")
plt.xlabel("Number of Animals")
plt.ylabel("Animal Age at Adoption")
plt.show()

# Second Hand Hound Animal Size histogram with 100 bars
plt.hist(shh.Size, 100)
plt.title("SHH Animal Size Historgram")
plt.xlabel("Animal Size (Weight in Pounds)")
plt.ylabel("Number of Animals")
plt.show()

#plt.hist(shh.isAdopted_90_Num, 100)
#plt.show()
```



Polynomial Regression, Animal Age at Adoption based on Size

6940 SHH - Data Prep - 02 Polynomial Regression.ipynb

File Edit View Insert Runtime Tools Help All changes saved

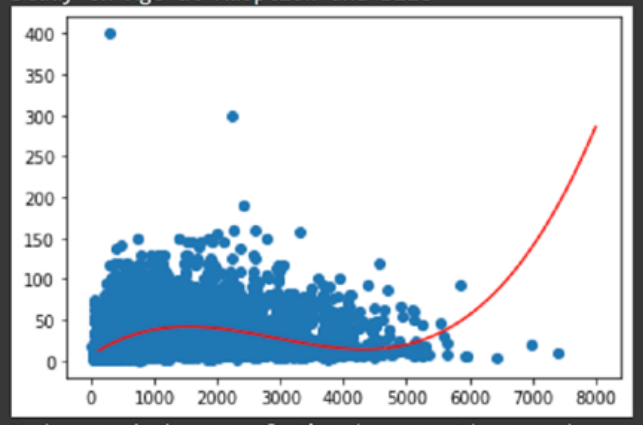
Comment Share

+ Code + Text

RAM Disk Editing

```
print('and determine if a polynomial regression is fit for prediction')
print('Values close to 0 mean no relationship, close to 1, correlated')

print('R-Squared')
print(r2_score(y, polynomial_regression_model(x)))

print('Predicted number of days to adoption based on size:')
predicted_size = polynomial_regression_model(40)
print(predicted_size)
```

```
Polynomial Regression
Study on Age at Adoption and Size
```




```
Understand the correlation between the x and y axes
and determine if a polynomial regression is fit for prediction
Values close to 0 mean no relationship, close to 1, correlated
R-Squared
0.257088123920721
Predicted number of days to adoption based on size:
9.172285509641188
```

0s completed at 5:51 PM

Decision Tree Classifier Gini – Confusion Matrix on Predicted SHH Adoption Timeframes

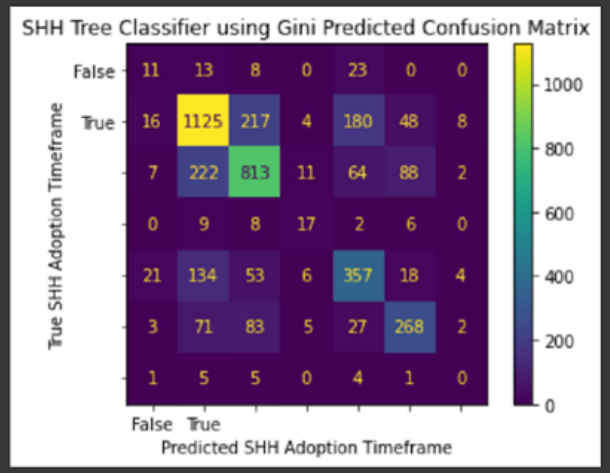
6940 SHH - Decision Tree Classifiers Gini & Entropy.ipynb

File Edit View Insert Runtime Tools Help Last saved at 2:15 PM

Comment Share

+ Code + Text Connect Editing

```
[ ]  1  confusion_matrix_gini = metrics.confusion_matrix(y_test, y_pred_gini)
     2
     3  # Convert the gini predicted table into a confusion matrix
     4  confusion_matrix_shh_adopt_gini = metrics.ConfusionMatrixDisplay(confusion_matrix = confusion_matrix_gini, display_labels = [False, True])
     5
     6  # Display the confusion matrix findings
     7  confusion_matrix_shh_adopt_gini.plot()
     8  plt.title("SHH Tree Classifier using Gini Predicted Confusion Matrix")
     9  plt.xlabel("Predicted SHH Adoption Timeframe")
    10  plt.ylabel("True SHH Adoption Timeframe")
    11  plt.show()
```




```
[ ]  1  # Plotting the decision boundary
```

GREAT LAKES
DATA & ANALYTICS
SUMMIT

Thank you for attending!

Hosted by: