Furiends Presentation 2

Yannan Niu, Yiting Gan, Sining Shen, Xiaojian Li

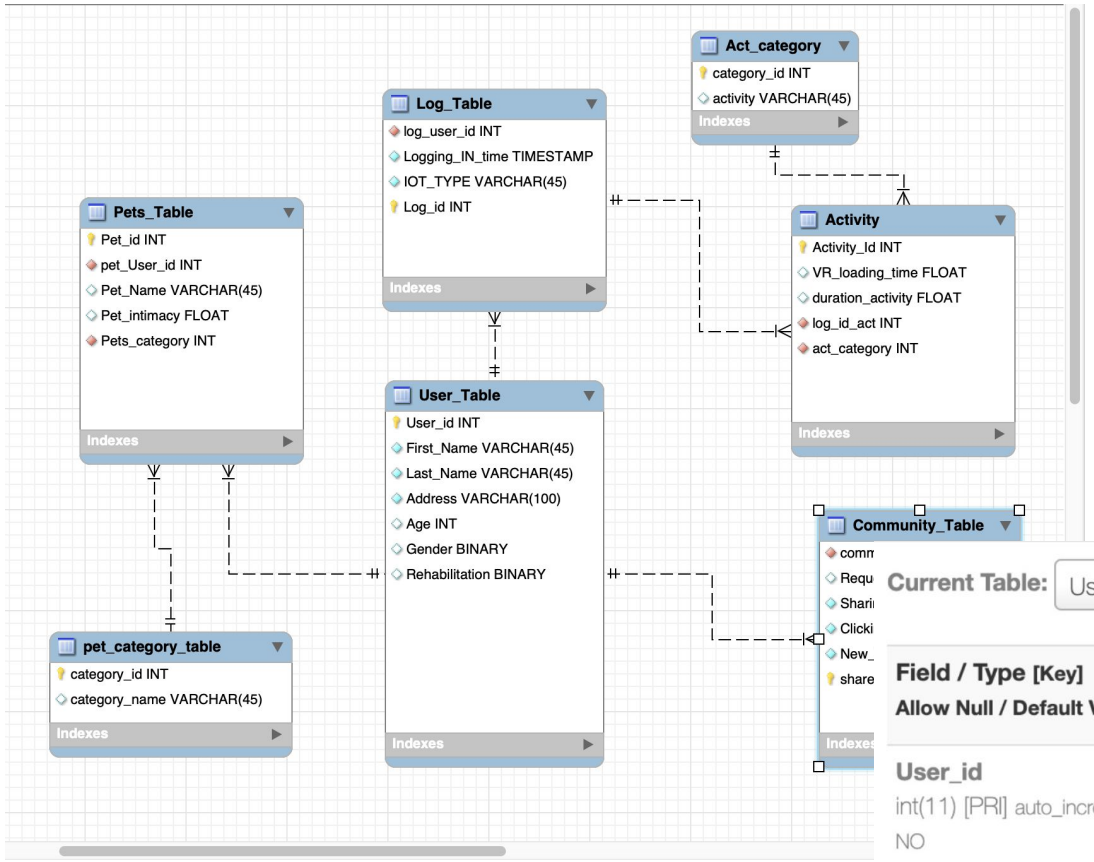
Data Gathered

Synthesis Data

- Bummer:
  NO history data from the company 😫

- Solution:
  1. Clarify all the variables/attributes we need in database, along with their type and constraints
  2. Search options for generating synthesis data
  3. Land on a website, where we input schema and requirements and it outputs randomized datasets

Data Info



Current Table: User_Table — 0 rows

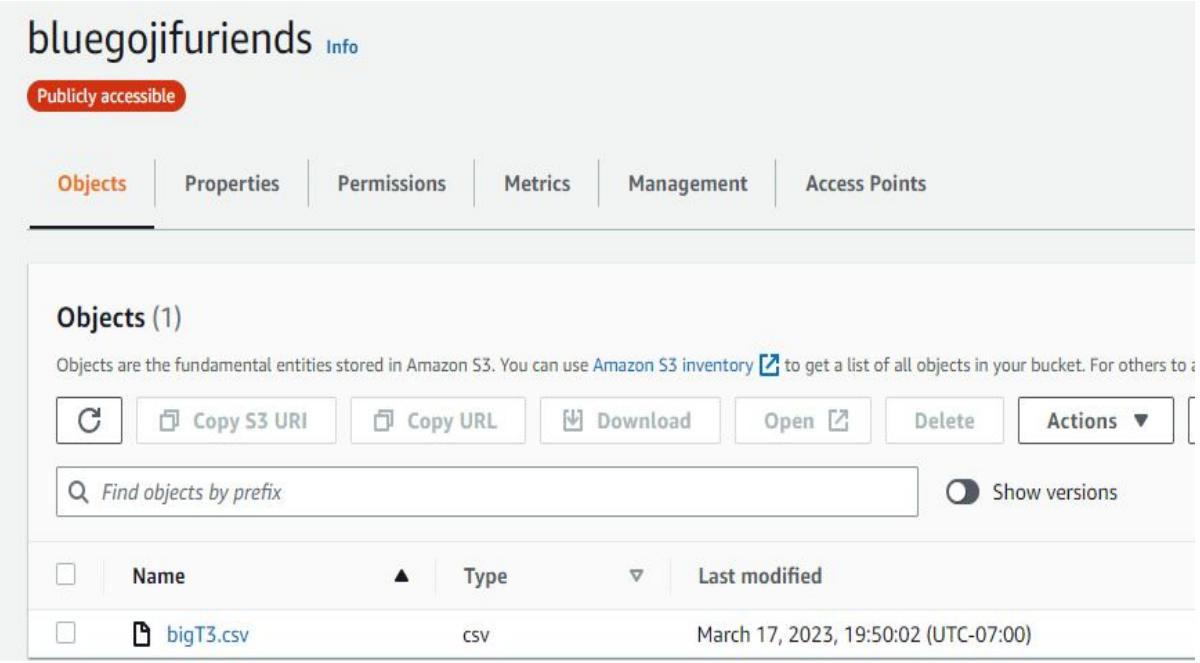
Field / Type [Key] / Allow Null / Default Value	Generate — Select type of data to be generated for every column	Parameters — Comma separated	Unique	Opt-nal
User_id int(11) [PRI] auto_increment | NO	*Autoincrement, generated by MYSQL* If current table contain Foreign Key(s), please ensure that table that refers contain records, otherwise generate data for referenced table first.			
First_Name varchar(45) | NO	firstName($gender = null|'male'|'female')// 'Maynard'		☐	☐
Last_Name varchar(45) | NO	lastName // 'Zulauf'		☐	☐
Address varchar(100) | NO	address // '8888 Cummings Vista Apt. 101, Susanbury, …		☐	☐
Age int(11) | YES	randomNumber($from, $to) // 39049	12,90	☐	☐
Gender binary(1) | YES	randomNumber($from, $to) // 39049	1,2	☐	☐
Rehabilitation binary(1) | YES	randomNumber($from, $to) // 39049	1,2	☐	☐

How We Resolve Data

Cloud Storage

Amazon S3

- Amazon S3 (Simple Storage Service) is a cloud-based object storage service provided by Amazon Web Services (AWS).
- It allows users to store and retrieve large amounts of data, including text, images, videos, and any other type of digital asset



Advantage of S3 Bucket

Advantage:

- Scalability: S3 is designed to be highly scalable and can handle virtually unlimited amounts of data.

- Durability: S3 provides high durability for stored data, ensuring that data remains available and retrievable even in the event of hardware failures or other issues.

- Accessibility: S3 provides a web-based interface for accessing stored data, making it easy to access and share data across multiple applications and platforms.

Target functions

```python
def normalize_series(series):
    min_val = series.min()
    max_val = series.max()
    normalized_series = series.apply(lambda x: (x - min_val) / (max_val - min_val))
    return normalized_series
```

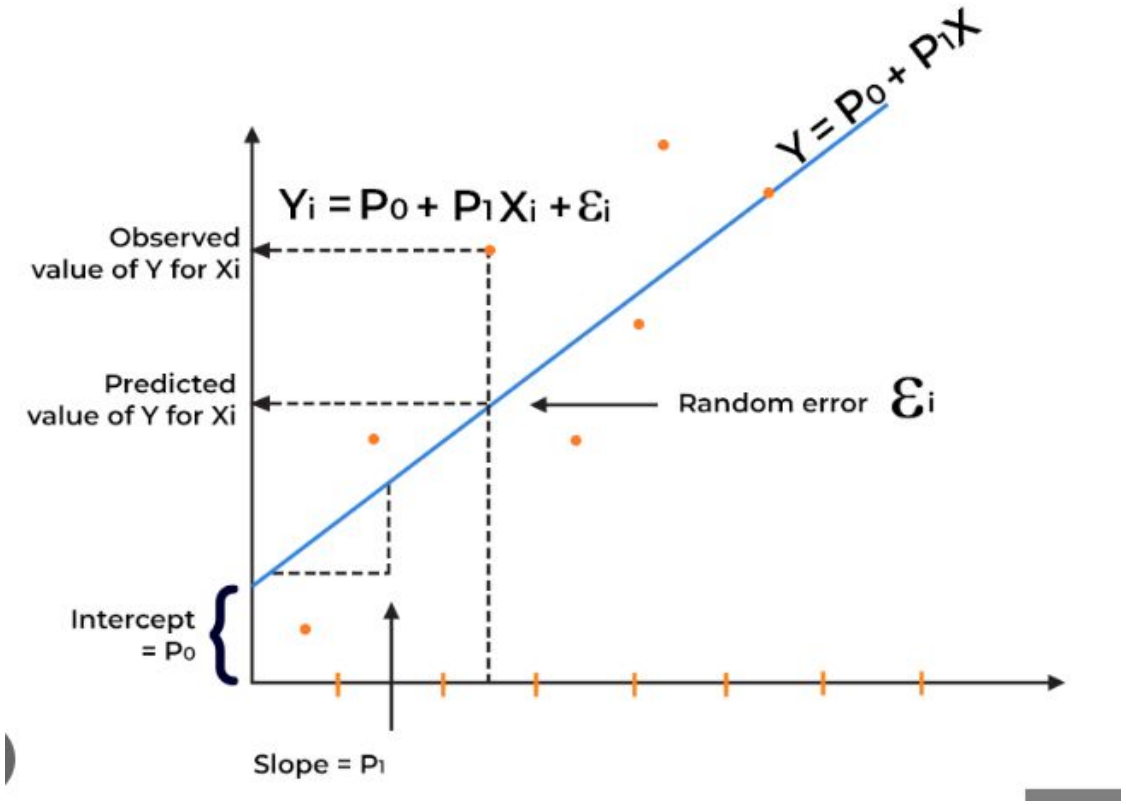
```python
def calculate_user_satisfaction(row):
    # Define weights for each metric
    # metric1: rehab, metric2, pet_intimacy, 3: category_name,4 log_count,5 com_count, 6: avg_loading_time,7 sum_duration
    weights = [0, 0.1, 0.1, 0.2, 0.1, 0.1, 0.4]
    metric1, metric2, metric3, metric4, metric5, metric6, metric7 = row[0],row[1],row[2],row[3],row[4],row[5],row[6]
    if metric1 == 1: # means it is for rehabilitation otherwise it is 0.2
        weights[0] = 0.2
    if metric3 == "Golden Retriever": # means it is for rehabilitation otherwise it is 0.2
        weights[2] = 0.5
    # Calculate the weighted average score
    weighted_score = (weights[0]*metric1 + weights[1]*metric2 + weights[2]*0.5 +
                      weights[3]*metric4 + weights[4]*metric5 + weights[5]*metric6+ weights[6]*metric7)

    # Return the user satisfaction score
    return weighted_score
```

Why Target functions

- Model evaluation: Having a target function enables you to measure the performance of the ML algorithm by comparing its predictions to the actual outcomes provided by the target function.
- Supervised learning: In supervised learning, the target function serves as a label or target variable, which guides the algorithm to learn the mapping between input and output effectively.
- Synthetic data generation: A target function helps in generating synthetic data with specific patterns, allowing you to simulate various scenarios and test the performance of your ML models under different conditions.
- Ground truth establishment: A target function provides a clear reference to the desired output, allowing the algorithm to learn the relationship between input features and the expected outcome.

Linear Regression Model

Linear Regression



Linear Regression Model

Assumption

1 Linearity

2 Constant Variance Homoscedasticity

3 No-Autocorrelation

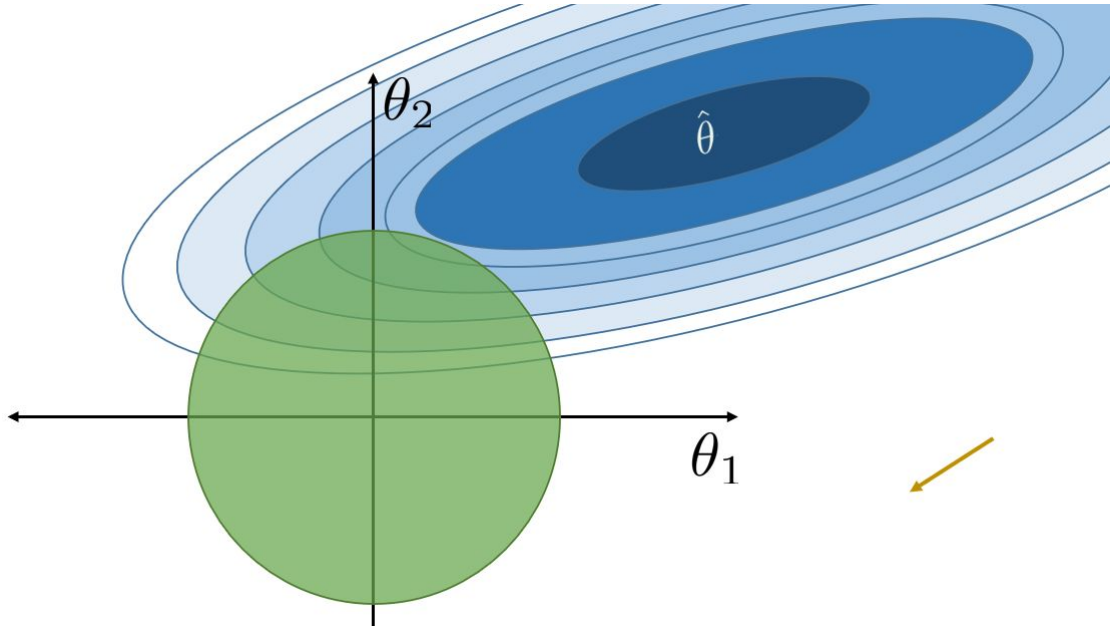
4 Independent samping

5 No collinearity

6 error term follows $N(0,\sigma^2)$

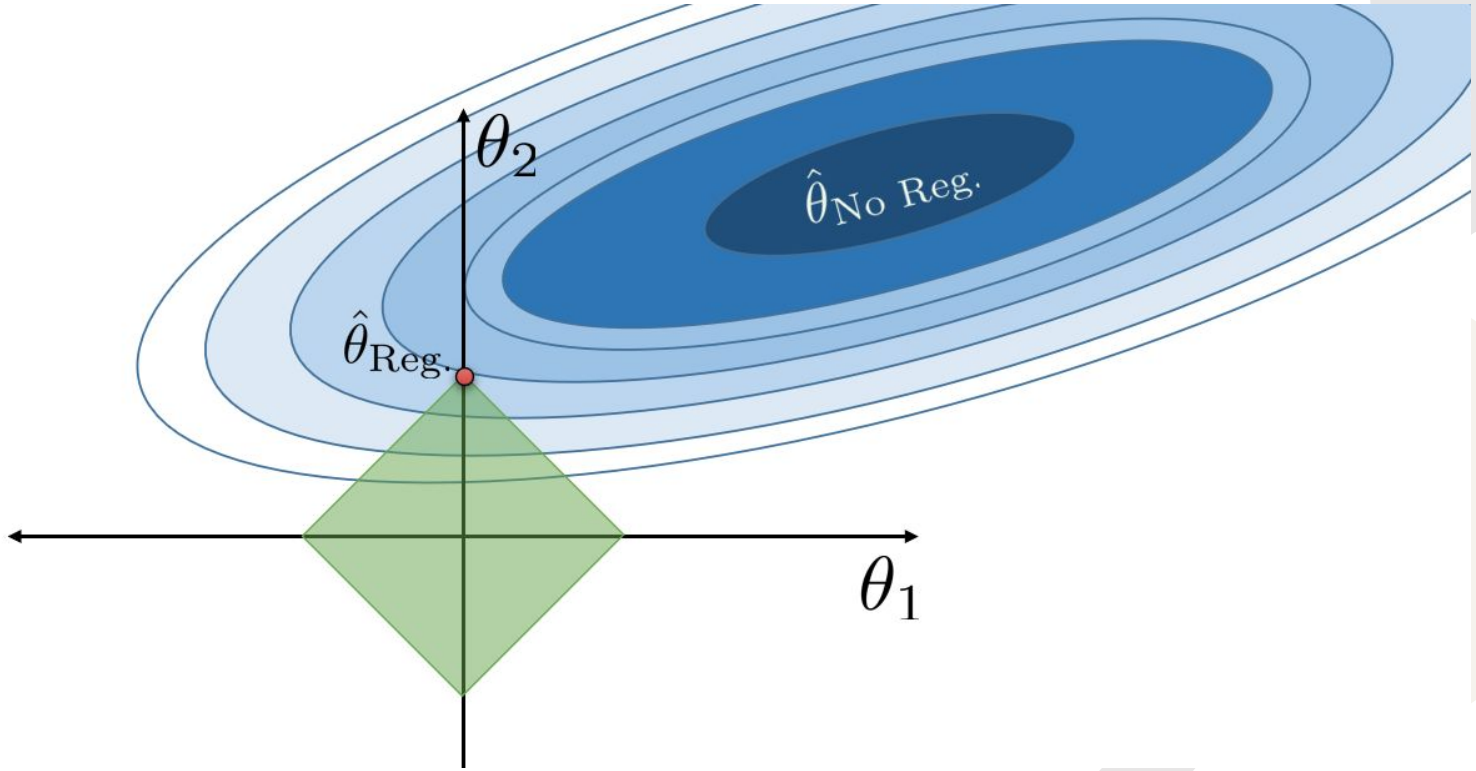
Ridge Regression Model (L2)

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - (\theta_0 + \theta_1\phi_{i,1} + \cdots + \theta_d\phi_{i,d}))^2 + \lambda\sum_{j=1}^{d}\theta_j^2$$



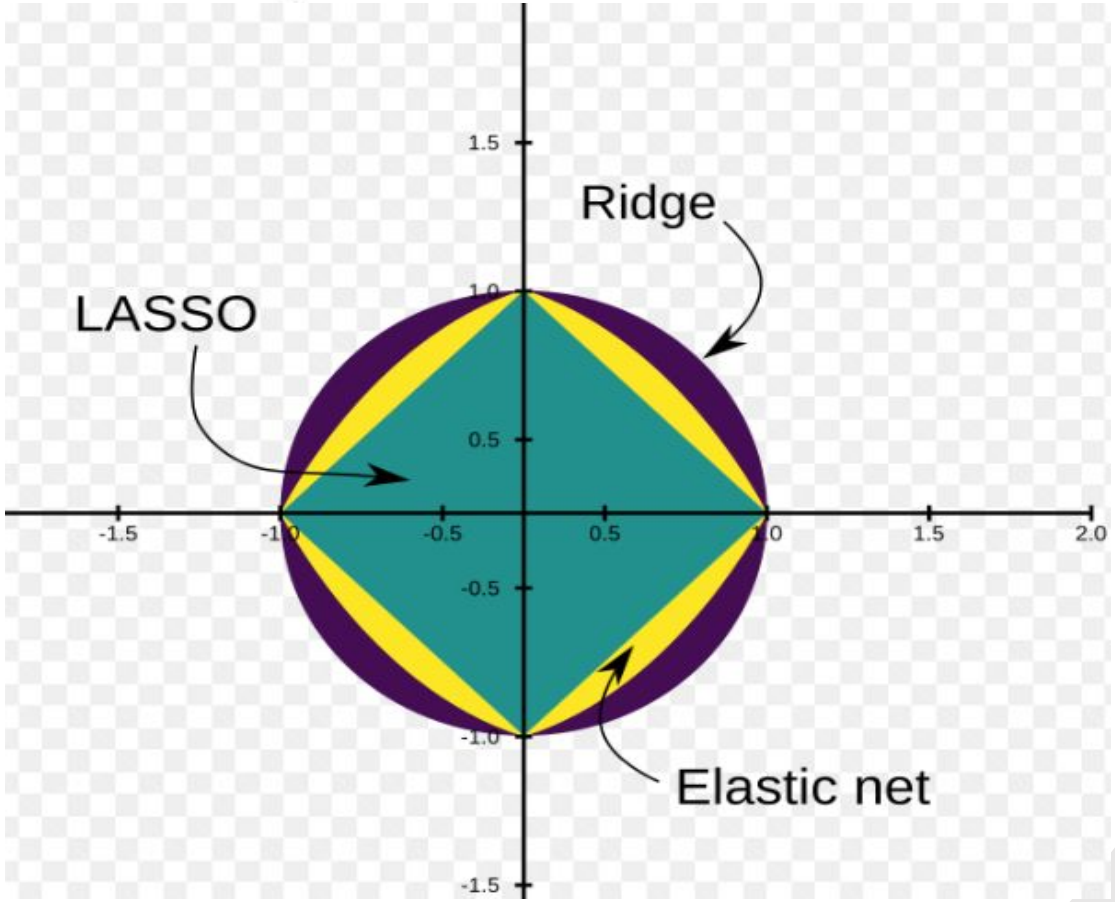
Lasso Regression Model(L1)

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - (\theta_0 + \theta_1\phi_{i,1} + \cdots + \theta_d\phi_{i,d}))^2 + \lambda\sum_{j=1}^{d}|\theta_j|$$



Elastic Net Regression (Hastie 2004 stanford University)

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^{N} \left(y_i - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda_1 \sum_{j=1}^{p} |\beta_j| + \lambda_2 \sum_{j=1}^{p} \beta_j^2 \right\}$$



Why Regularization ?

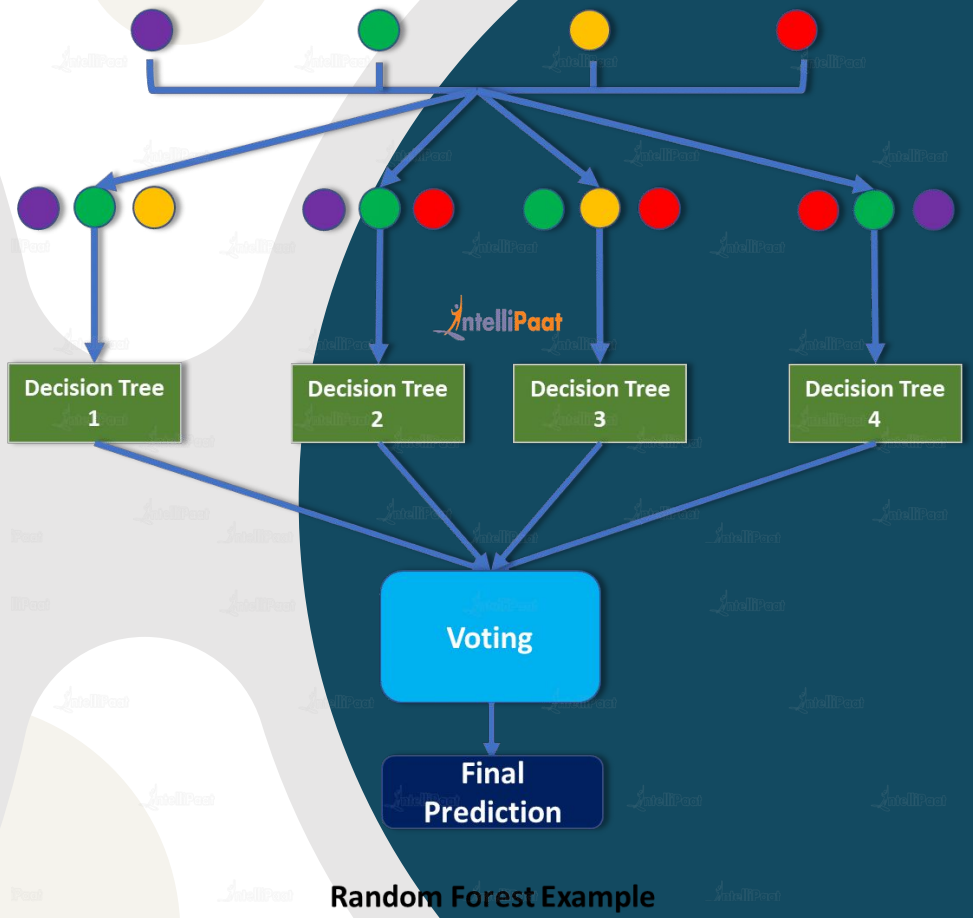
- Decrease Overfitting
- More generalization of models
- Better MSE. Meaning better accuracy
- In our case, we could have more opportunities to explore the more factors, which may play a significant role in determining user satisfaction. Otherwise, we may overestimate the importance of certain factors and neglect the others.

Random Forest model

- Logic behind the model
  1. Randomly select a subset of data from the training set.
  2. Create a decision tree using the selected data subset by selecting the “best split” at each node based on a random subset of features.
  1. Repeat steps 1 and 2 to create a forest of decision trees.
  2. Predict new data points by aggregating the predictions of all decision trees in the forest.

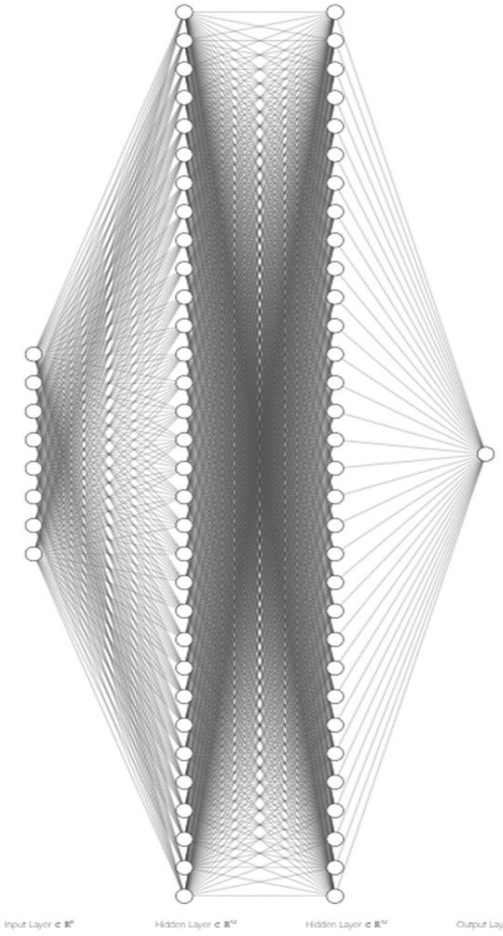
- Parameters Modification
  1. the number of trees
  2. maximum depth
  3. the number of features to consider at each split

- Evaluation
  1. mse: 0.07694 -> 0.0759
  2. Best parameters: {'max_depth': 20, 'min_samples_split': 2, 'n_estimators': 100}



Random Forest Example

Neural Network



Three layers: 32, 32, 1

Columns used in independent variables: "Age", "Gender", "Rehabilitation", "Pet_intimacy", "log_count", "com_count", "avg_loading", "sum_duration"

Dependent Variable: User Satisfaction

Train loss(mse): 0.0872

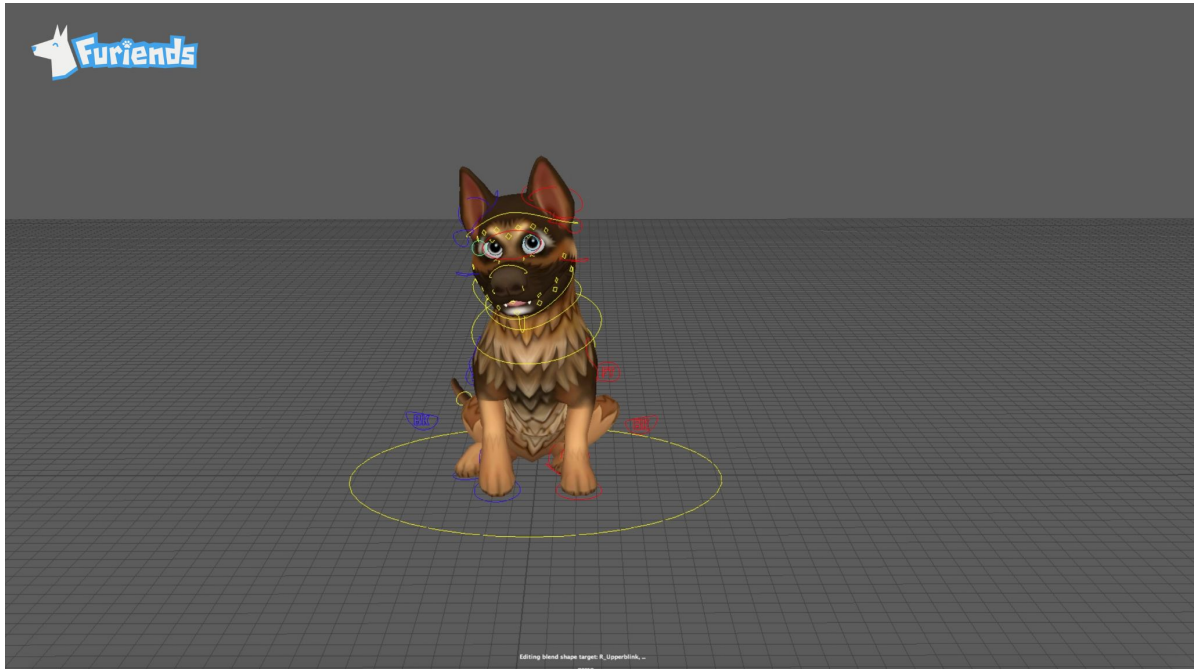
Validation loss(mse): 0.1229

Test loss(mse): 0.1356

Our Future Plan

Exploration and estimation of some possibilities

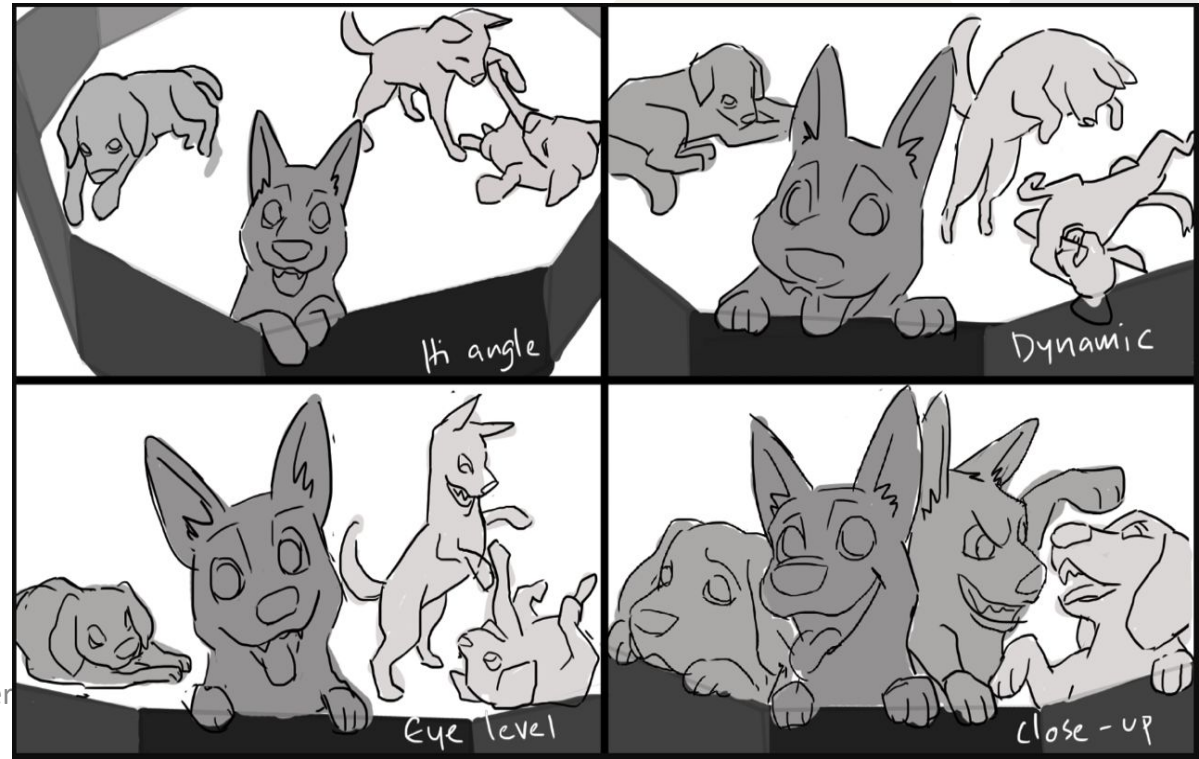
Possible Direction For Improvement



- Database
- Model
- Game quality
- User experience

What the future team can do

- Expand and enrich the database, upgrade more data types, and refine existing data types.
- Optimize machine learning models to increase accuracy and efficiency.
- Conduct more real user tests to obtain more data to further train the model.
- Improve the quality of the puppy model and the interaction between humans and dogs to provide a better user experience and enable more rehabilitation training functions.



Thank You!