

Pet Expense Management

A Data Science Endeavour

Pet ownership in Germany has increased by over 30% since the pandemic and currently lies at over 15.7 million cats and 10.7 million dogs (2020, Zenz-Spitzenweg). Dogs are the second most popular pets, but dog owners spend far more than any other category of pet owner.

Objectives

- Determine the key criteria most correlated with dog ownership expenses.
- Identify risk factors associated with higher costs for individual dogs.
- Explore KPIs that could be used as early warning indicators for avoidable expenses.

Table of Contents

Introduction.....	3
Context.....	4
Foundation Dataset.....	4
Data Limitations.....	5
Data Exploration.....	6
Variable Selection.....	9
Data Concept.....	10
Machine Learning Results.....	12
Feature Engineering.....	12
Model Selection.....	13
Model Interpretation.....	13
Random Forest for Classification.....	14
Cross validation on filtered dataset (selected feature importance).....	16
Performance after Regularization.....	17
Cross-validation on model regularized.....	17
SHAP Chart (Random Forest).....	18
LIME (on Random Forest).....	18
Feature Perturbation (sensitivity analysis).....	20
Decision Tree with Regularization.....	21
Hyperparameter Tuning with RandomizedSearchCV (second-order regularization):	22
Random Forest Performance.....	23
Decision Tree Performance.....	23
Comparative Analysis.....	23
Random Forest for Regression.....	23
Random Forest before dimension reduction.....	24
Feature performance before dimension reduction (fig.1).....	24
Random Forest after dimension reduction.....	25
Feature performance after dimension reduction (fig.2).....	25
Binarizing the Data.....	27
Classification Reports.....	28
Classification Model Performance.....	31
K-Nearest Neighbors.....	32
Splitting the Dataset.....	32
Behaviour Only (with Breed).....	32
LightGBM (with Breed) Behaviour Feature Importance Ranking.....	33
Behaviour Only (without Breed).....	33
LightGBM (without Breed) Behaviour Feature Importance Ranking.....	34
Ailments Only (with Breed).....	34

LightGBM (with Breed) Ailments Feature Importance Ranking.....	35
Ailments Only (without Breed).....	35
LightGBM (without Breed) Ailments Feature Importance Ranking.....	36
Support Vector Machine.....	36
Change of Target Variable.....	36
Key Takeaways.....	37
Conclusion.....	38
Bibliography.....	39
Appendix.....	40

Introduction

This data science project has its origins in the business case of Paw à Peau (a pet tech company founded by Natasha Azza). Paw à Peau is building a platform to convert animal health and behavioural data into actionable insights for pet owners. The project's challenge was thus to translate the benefits of collecting data into financial terms. The idea was to create an alert system to integrate with a planned expense management feature on the platform.

We chose to focus on dog owners, as they are the type of pet owner who spends the most on their pets. We initially aimed to determine the real cost of dog ownership in the European market based on variables such as breed, age, lifespan, health issues, and behavioural traits. This was to be a prerequisite for exploring KPIs as early warning indicators for avoidable expenses.

The team members consist of Natasha Azza (startup veteran and former Head of Product) and Veronica Benzi (experienced data analyst with a PhD in Biology). Both team members are avid animal lovers and thus shared a special interest in this somewhat niche topic. The division of work was made according to skill set, with Natasha focusing primarily on the business side of things and bringing domain knowledge to the table, and Veronica acting as the technical lead. The project consisted of three phases:

- Data exploration
- Data modelling
- Presentation

The most time-consuming part of the project was data cleaning. As we were using data from various sources, including our own primary calculations, we were faced with challenges such as missing data, dirty data (e.g. from another region, where certain transformations had to be made such as currency conversion and cost normalization) and biased datasets. Data exploration did not bring us much value in terms of choosing the machine learning model itself or in framing our business problem, as we already had extensively researched the domain prior to this project.

We initially categorized the machine learning problem of the project as regression, but later moved to classification due to much of the data originating from our own work in feature engineering. The task of the project relates to anomaly detection within a dataset. We decided to use accuracy as the main performance metric to compare our models, due to the tendency towards overfitting (most likely caused by the nature of the data).

We tested four different algorithms: Random Forest, Decision Tree, K-Nearest Neighbors and Support Vector Machine. We further investigated the models utilizing Grid Search, Cross Validation, and Boosting, among other techniques. We also engaged in extensive feature engineering and tried different target variables, in order to bring the accuracy of the model down to address our main issue of overfitting. For interpretability, we used SHAP and LIME techniques primarily.

Context

Since the pandemic, pet ownership has skyrocketed and all segments of the pet industry (from retail to pet services and emerging technologies) are currently booming in the United States. Europe is following closely behind, with the UK, Germany and France as the biggest and fastest growing markets. Germany is the EU's largest economy but has been hit hard by rising costs stemming from energy prices, inflation, and financial side effects from ongoing regional wars. Two steep increases in the national veterinary tariffs within the last three mean pet owners (47% of Germany's population) are under financial pressure.

Dog ownership expenses can be classified into the following categories:

- Food
- Accessories
- Medical
- Grooming
- Training
- Funerary
- Tax (particular to Germany)
- Insurance
- Pet sitting

The costs vary for each individual dog and depend on a variety of factors such as breed, size, age etc. This is reflected in the fact that larger pets consume more, are more destructive, cost more to treat and are faced with higher insurance premiums.

Foundation Dataset

The foundation dataset "Best In Show" is open source and available on Kaggle. It was chosen due to the vast number of variables (69) and the inclusion of cost data (albeit related to the US market).

```
[ ] best_cleaned.head()
```

	Dog breed	category	LIFETIME COST, \$	2 LONGEVITY	4b food costs per year, US\$	5a grooming required	size category	weight (kg)	shoulder height (cm)	intelligence category
0	Additional info	American Kennel Club group	NaN	years, weighted average - see note	NaN	once every.. 1=day, 2=week 3=few weeks	NaN	mean of 'ideal' range for breed, mean of dogs ...	NaN	NaN
1	Border Collie	herding	\$20,143	12.52	\$324	2	medium	no data	51	Brightest
2	Border Terrier	terrier	\$22,638	14.00	\$324	2	small	6	no data	Above average
3	Brittany	sporting	\$22,589	12.92	\$466	2	medium	16	48	Excellent
4	Cairn Terrier	terrier	\$21,992	13.84	\$324	2	small	6	25	Above average

Further datasets were chosen on the basis of complementary variables which could enhance the relationships between variables in our foundation dataset. Variable connections are highlighted in a preliminary schema of the selected variables from each dataset (see cover link to Miro board).

The biggest challenge is combining data from many sources and deciding which variables to keep or drop. This is due to variance in data quality and data availability. For example, longevity data is abundant with 440k rows. Bite statistics, however, is limited although the data is useful as an enhancement. After cleaning the data and first explorations, we shall finalize the selection.

Pet sitting expenses will not be included in the overall cost calculation due to the irregularity of this variable. Not all dog owners use professional boarding kennels and instead may opt to use private services or the help of friends and family, meaning costs may be non-existent (and at best unpredictable). Furthermore, the time spent away from pets varies from one household to another.

Data Limitations

Two new columns were added: veterinary costs and training probability. Veterinary costs make up a large amount of pet owner expenses due to the low incidence of insurance ($\pm 5\%$), however these costs were not included in our foundation data set. As there is an abundance of recent economic studies on pet ownership in Germany, financial data from the results of the Ohr and Zenz-Spitzweg studies will be used to enrich our dataset.

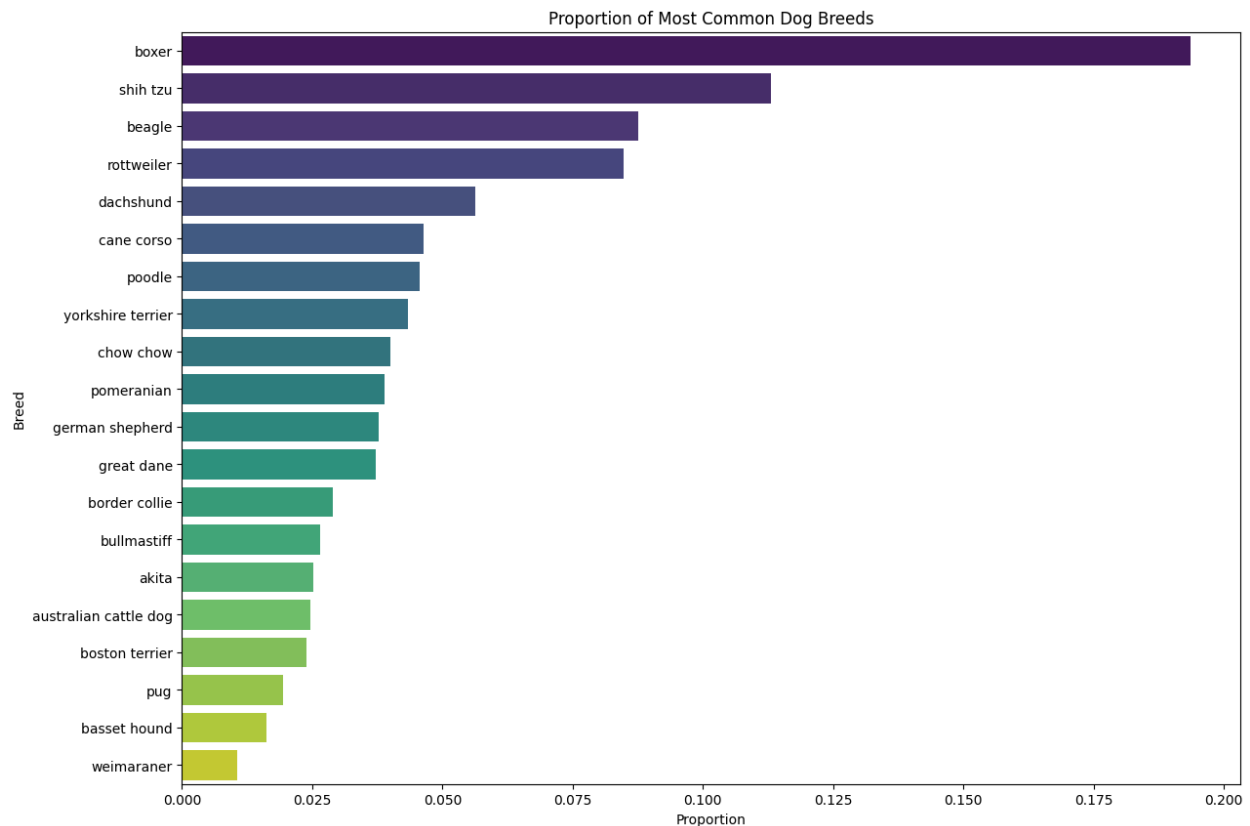
Training course costs vary greatly by institute and course type, however the hourly fee range of trainers is readily available online. The probability that a dog needs training can be extracted from the dataset through variable correlation (breed, intelligence, size, age etc.). We would thus attempt to predict training needs as a score. Depending on available data, it may be possible to also predict real costs based on the probability that a dog needs training, average trainer fee, and average duration of training according to lifestage or behavioural problem.

Retail financial data is readily available for the US and UK markets, but not for continental Europe. It may be necessary to approximate retail expenses based on the differences in cost of living, where local data is not readily available. Some of the data is provided with a calculation based on weekly, monthly, or yearly expenses. These sums will need to be normalized in order to properly explore the data.

Grooming costs depend on both size and coat type. Although able to be calculated based on local groomer fees, pet owners often forgo professional services in favour of a cut at home or simply letting the dog grow its coat out. As such, grooming (if kept in the final selection) should rather be converted into a categorical variable.

Data Exploration

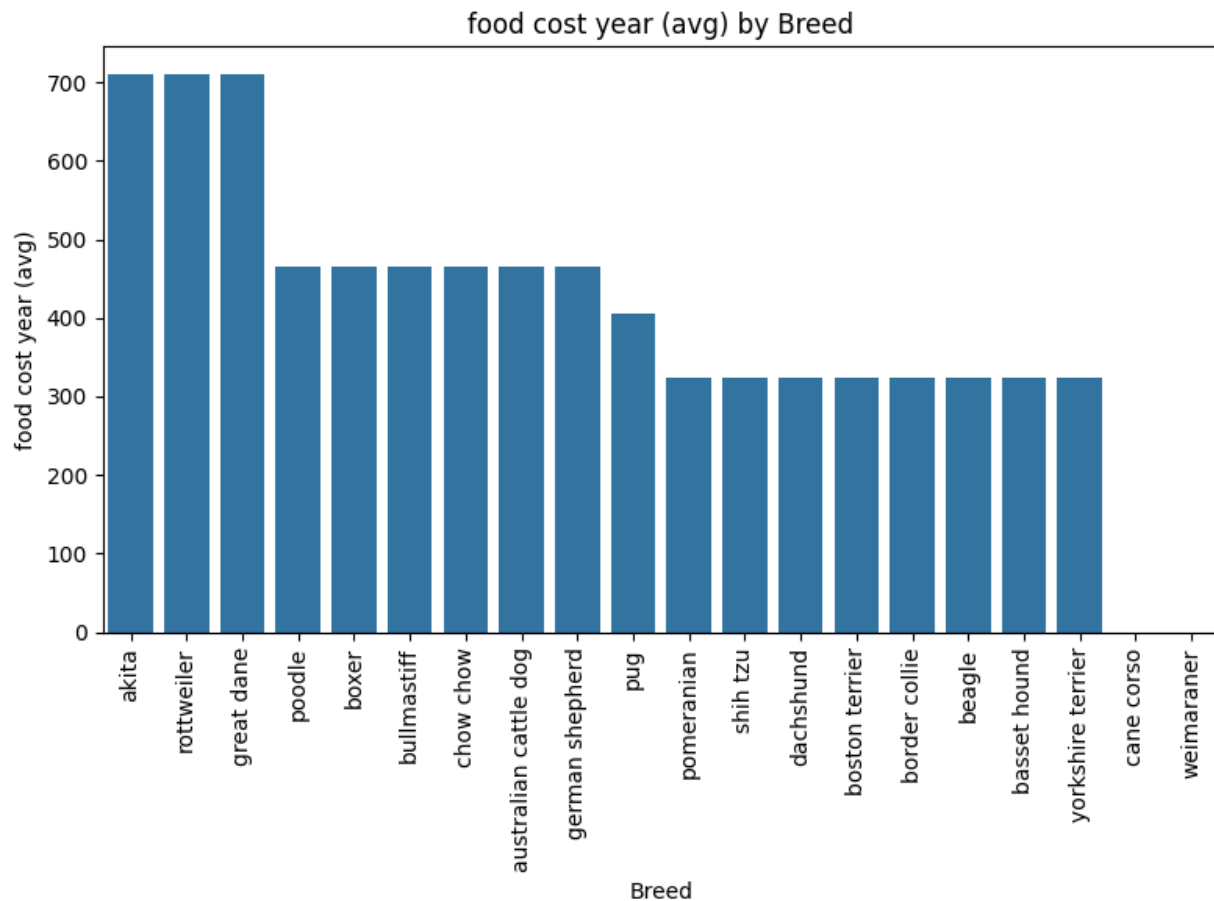
Some breeds had to be excluded from the final dataset due to a lack of data. We made sure to include the most popular breeds in Germany by combining additional health-focused datasets. To account for dogs that do not fit within this selection, we created three new rows to represent mixed breeds of the three different size categories, and used the mean and mode data to fill out each variable.



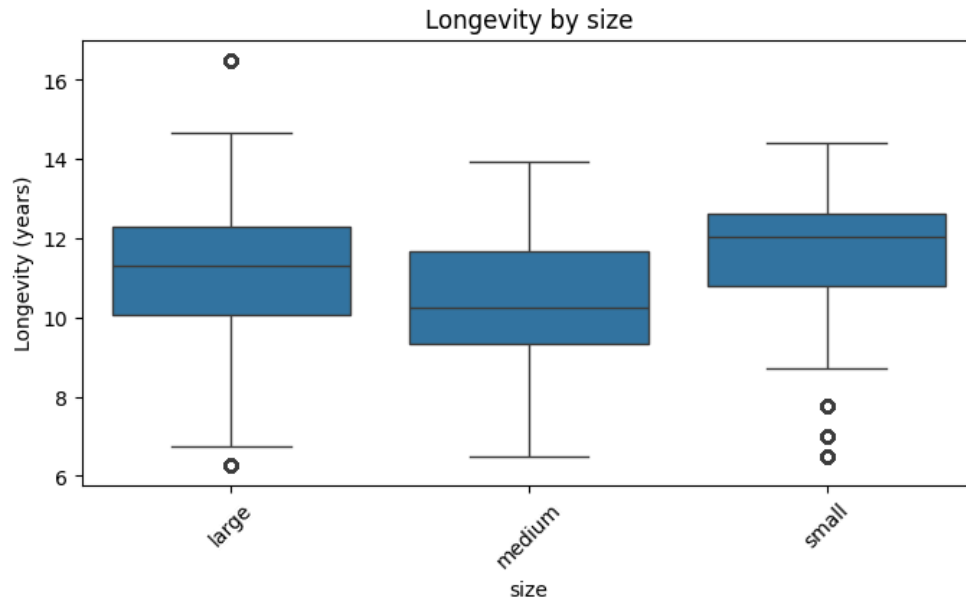
After joining multiple data sets, however, it became clear that we had unwittingly created a strong bias for certain breeds due to their overrepresentation in some of the health datasets we used.

At this point, our dataset was extremely large at over 1 million rows. We thus decided to limit the number of dogs that could be represented by any one breed to 1500, which were chosen at random. As there are many dog breeds in existence, even this limitation still left us with approximately 200 000 rows in the final dataset.

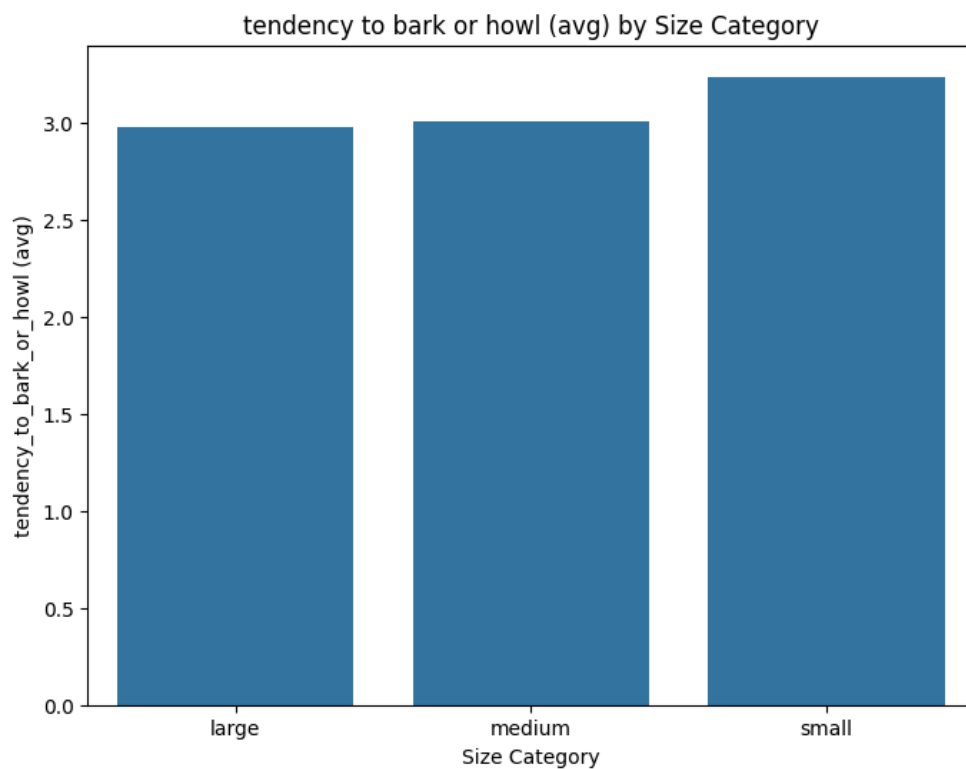
Factors which greatly influence cost include longevity (long life means more accumulated costs), breed (predisposition for certain health and behavioural traits) and size (directly related to the volume of food intake). The existing calculation within the dataset of food costs by dog size alone may not be accurate due to certain breeds having different activity levels.



We discovered that the dataset needed extensive cleaning following exploration of correlation between longevity and size. It is well documented in scientific literature that smaller dogs tend to live longer than larger dogs. This, however, was not represented in the graph and there was no plausible explanation as for why medium sized dogs would be shorter lived than larger breeds.



Another interesting correlation to explore is different temperament traits which correlate with the tendency to bark or howl. Noise and destruction within the home are some of the main reasons why dog owners may seek out the help of a trainer, or failing that, abandon their pet. Behaviour-related variables are thus important to understand when it comes to the potential lifetime costs of owning a dog.



Variable Selection

Pre-cleaning Selection

Our pre-cleaning variable shortlist:

'Breed', 'lifetime_cost', 'longevity', 'food_cost_year', 'grooming required ', 'size_category', 'weight_kg', 'shoulder_height_cm', 'intelligence_category', 'avg_food_per_week_£', 'food_per_week_\$', 'genetic_ailments', 'category', 'Specie', 'Gender', 'where_bitten', 'sensitivity_level', 'tolerates_being_alone', 'tolerates_cold_weather', 'tolerates_hot_weather', 'incredibly_kifriendly_dogs', 'dog_friendly', 'friendly_towardstrangers', 'easy_to_groom', 'potential_for_mouthiness', 'prey_drive', 'tendency_to_bark_or_howl', 'wanderlust_potential', 'exercise_needs', 'energy_level', 'adapts_well_to_apartment_living'

We began with many variables and datasets due to the complexity of the pricing data involved in pet ownership. Food, accessory, and veterinary costs were the only readily available data (the latter, as part of a study and not from a dataset). The primary target variable was lifetime_cost of the individual dog. We aimed to generate multipliers for training probability, insurance premiums, and other variables to the available mean data. We were not yet certain whether the cost calculator will be limited to dogs only, or also include cats (for which data is abundant, and there is little variance between breeds).

There was a correlation between certain behavioural elements, as well as between size and breed, size and longevity, and size and cost. What we were uncertain of, and what required further exploration was the extent to which these variables were related and could be used to predict lifetime cost. Another factor to take into account is whether we want to only predict the cost over the entire lifetime of a dog with certain characteristics (such as breed, size, sex etc.), or also predict the remaining costs based on the current age of the dog (and multiplied by other factors).

Post-cleaning

New cost variables were added, including veterinary_cost, insurance_cost, tax_cost, ailment_cost, training_probability etc. due to the inaccuracy of predicting lifetime cost based solely on food and accessory expenditure.

The shelter datasets were dropped due to an insufficient number of rows (it made no sense to combine datasets with such discrepancy). Nevertheless, we have kept the numerous datasets with the goal of using them to enrich or verify the data in our foundation dataset. We also dropped the following superfluous variables: 'weight_kg', 'shoulder_height_cm', 'avg_food_per_week_£', 'food_per_week_\$', 'Specie', 'where_bitten', 'adaptability'

Leaving:

'Breed', 'breed_group', 'longevity', 'size_category', 'intelligence_category', 'genetic_ailments', 'neutered', 'sex', 'grooming_required', 'food_cost_year', 'lifetime_cost', 'sensitivity_level', 'tolerates_being_alone', 'tolerates_cold_weather', 'tolerates_hot_weather', 'incredibly_kifriendly_dogs', 'dog_friendly', 'friendly_towardstrangers', 'potential_for_mouthiness', 'prey_drive', 'tendency_to_bark_or_howl', 'wanderlust_potential', 'exercise_needs', 'energy_level', 'adapts_well_to_apartment_living', 'bite_statistics'

([Variable Index](#) - Data Audit - see detailed description of variables here)

Data Concept

A significant amount of transformation and normalization will need to be done before modeling can begin. This has become clear during the cleaning and variable selection process. We had begun with the idea of doing regression modelling to predict total lifetime costs of owning a dog with certain characteristics.

The screenshot shows a web form titled "Dog Ownership Cost Calculator". At the top, there are navigation icons (back, forward, refresh) and a close button (X). Below the title, a descriptive text states: "This calculator will provide a prediction of the costs you can expect from dog ownership according to your individual dog's characteristics. Adjust the criteria below and click 'predict'." The form contains several input fields and radio button groups. On the left, there is a text input for "Enter your dog's name", a dropdown for "Choose Dog Breed", a "Sex" section with "Female" (selected) and "Male" options, and a "Sterilized" section with "Yes" (selected) and "No" options. On the right, there is a dropdown for "Choose Age", a "Chronic diseases (incl. allergies)" section with options "0" (selected), "1", "2", and "3 or more", and a "Behavioural issues" section with "Yes" (selected) and "No" options. At the bottom center is a blue "Predict" button. The "miro" logo is in the bottom right corner.

Dog Ownership Cost Calculator

This calculator will provide a prediction of the costs you can expect from dog ownership according to your individual dog's characteristics. Adjust the criteria below and click "predict".

Enter your dog's name

Choose Age

Choose Dog Breed

Sex

☒ Female

☐ Male

Sterilized

☒ Yes

☐ No

Chronic diseases (incl. allergies)

☒ 0

☐ 1

☐ 2

☐ 3 or more

Behavioural issues

☒ Yes

☐ No

Predict

miro

←

→

↺

×

Dog Ownership Cost Results

Below is the lifetime cost estimate according to [Ben]'s breed average life expectancy of [age] years. It includes one-time set up and funerary costs.

€10,988

The following is an individualized yearly cost breakdown for "Ben", according to your input.

Food Expenses	€240	Total yearly costs €2577
Medical Expenses	€277	
Dog Tax	€180	
Medical Insurance	€600	
Training Costs	€600	
Grooming Costs	€600	
Accessory Costs	€80	

Back

miro

However, as it became clear that we had extensive cleaning and manual enrichment to do, it made more sense to change our strategy to a classification model. We will thus still roughly calculate lifetime costs, however we will present the user with an expense category and cost ranges.

←

→

↺

×

Dog Expense Risk Results

Below is the lifetime cost estimate according to [Ben]'s breed size and an average life expectancy of [age] years. It includes one-time set up and funerary costs.

€10,988

The following is an individualized yearly increased cost probability for "Ben", according to his breed and your input.

Food Expenses	2.0x	Overall Cost Score high (€2577- €4000)
Medical Expenses	5.0x	
Dog Tax	1.0x	
Medical Insurance	3.0x	
Training Costs	4.0x	
Grooming Costs	-1.0x	
Accessory Costs	1.0x	

Back

miro

Machine Learning Results

In this phase of our project, we were to choose a machine learning algorithm to train and tune a model to predict the categories of yearly expenses. We created five categories, labelled 'lowest', 'low', 'medium', 'high' and 'highest', representing yearly expenses binned by calculated quartiles.

Feature Engineering

The feature engineering process took a lot of time due to missing, dirty, or incorrectly formatted data. We combined many different datasets (as can be seen in the references section) and therefore we had to make decisions about which format to prefer over the others and subsequently transform the complementary datasets accordingly. Additionally, we enriched the costs using figures from scientific literature and our own primary research.

We standardized the data by ensuring certain thresholds for breed inclusion in the dataset, this included the minimum number of dogs which belonged to a given breed population, as well as the acceptable amount of missing key data. For example, dog breeds missing our key behaviour traits were dropped from the dataset. We also chose to represent mixed breeds by using the mode for missing data, i.e. behaviour traits, as well as figures presented in literature such as longevity, reduced insurance cost, and the decision to not associate the population with specific genetic ailments.

We analyzed the dataset with one hot encoded transformed 'breed' data, as well as without 'breed' data, and with breed labels encoded simply with numerics. Finally, we realized that 'breed' as a variable was not essential and that representation via 'breed_group' was sufficient. The foundation dataset had provided categorization based on the American Kennel Club, but we later changed this to the categorization provided by the Fédération Cynologique Internationale (an umbrella organization governing pedigree dogs in most of regions of the world) as these were more numerous, purpose-specific, and more relevant to Germany.

We dropped the following variables, due to irrelevance, better representation through other variables, or insufficient data:

'neutered', 'grooming_required', 'food_cost_year', 'lifetime_cost',
'adapts_well_to_apartment_living', 'bite_statistics'

We also expanded 'genetic_ailments' through one hot encoding, in order to have a more detailed breakdown of the costs. Following the calculations of individual genetic ailments and common training costs (local cost data in Berlin was used) to address certain behavioural problems, we removed all cost data apart from 'yearly_final_costs', which we used as our target.

Model Selection

After realizing that many individual cost parameters were missing from our dataset, we decided to change our approach away from regression and instead to classification. We tested various classification machine learning algorithms:

- Random Forest
- Decision Tree
- Boosting
- K-Nearest Neighbour
- Support Vector Machine

Our first step was preparing two common datasets, *combined_data.csv* (with breeds and ailments encoded) and *nobreed_data.csv* (without 'breed', only 'breed_cat'). We then separately tested two models each, along with various techniques such as Pruning and LIME. Additionally, various types of feature importance analyses were performed to determine the most important variables for the model's calculation. We selected Random Forest as our final model for the completion of the project due to its performance in comparison to the other models.

Model Interpretation

In our analysis, we aimed to evaluate the performance of our Random Forest model using both classification and regression approaches on the same dataset. This dual approach was undertaken for the following reasons:

Nature of the Target Variable:

Our dataset originally contained a continuous target variable. To effectively assess the model's performance, we initially discretized this continuous target into categorical bins. This allowed us to use a classification model to evaluate how well the target could be segmented into predefined categories. Simultaneously, retaining the continuous nature of the target allowed us to use regression models to explore how well we can predict the target as a continuous outcome.

By testing both approaches, we could identify which model provides a better fit for the dataset—and whether the target variable was best predicted as discrete classes or as a continuous value.

Random Forest for Classification

Random Forest classification models can capture non-linear relationships and interactions between features and the target variable, allowing us to evaluate the effectiveness of classifying the target into distinct categories.

One strategy we employed to standardize the data was balancing the samples of different cost categories, so that there was no model bias due to an overabundance of any given demographic. The expense classes were thus reduced to 4, as opposed to the original 5 ('lowest', 'low', 'medium', 'high').

We analyzed feature importance with Random forest before and after dimension reduction with PCA. And evaluate the model performance for both cases (with and without PCA)

Before PCA:

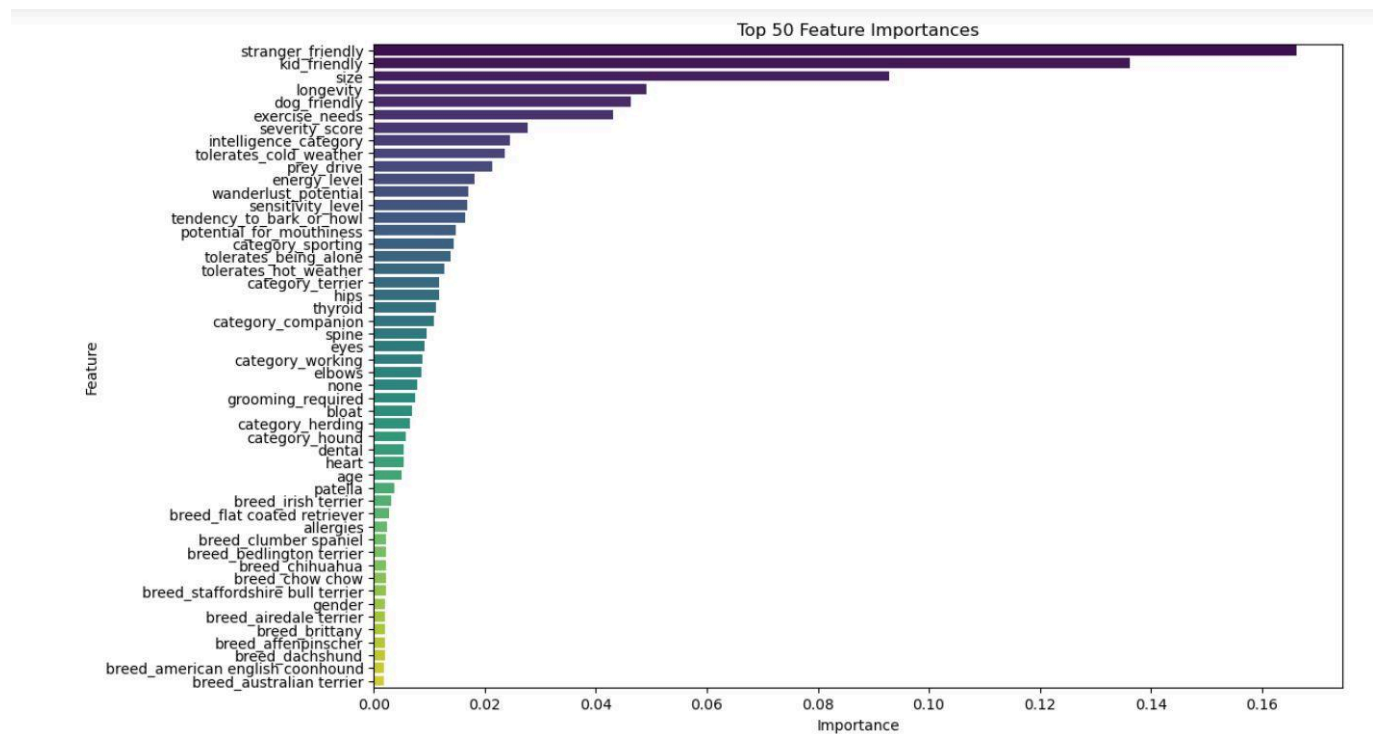
Accuracy before PCA: 1.00

Classification Report before PCA:

	precision	recall	f1-score	support
High	1.00	1.00	1.00	9339
Low	1.00	1.00	1.00	9726
Lowest	1.00	1.00	1.00	9416
Medium	1.00	1.00	1.00	9619
accuracy			1.00	38100
macro avg	1.00	1.00	1.00	38100
weighted avg	1.00	1.00	1.00	38100

Confusion Matrix before PCA:

```
[[9333  0  0  6]
 [  0 9723  0  3]
 [  0  2 9414  0]
 [  5 17  1 9596]]
```



After PCA:

Accuracy after PCA: 1.00

Classification Report after PCA:

	precision	recall	f1-score	support
High	1.00	1.00	1.00	9339
Low	1.00	1.00	1.00	9726
Lowest	1.00	1.00	1.00	9416
Medium	1.00	1.00	1.00	9619
accuracy			1.00	38100
macro avg	1.00	1.00	1.00	38100
weighted avg	1.00	1.00	1.00	38100

Confusion Matrix after PCA:

```
[[9339  0  0  0]
 [  0 9726  0  0]
 [  0  0 9416  0]
 [  0  3  0 9616]]
```

Using the Random Forest algorithm, we tried removing the variables that had little relevance - defined as ≤ 0.01 importance and re-testing the model's performance. The features of low importance which were removed included 'breed', some genetic ailments and 'grooming_required'.

Breed was already sufficiently represented by breed groups, which broadly categorized the dogs by purpose, for example 'category_working' includes breeds such as *Giant Schnauzer*, *Dobermann*, *German Shepherd*, *Malinois* and others which are typically used by police or security. Breed groups tend to share similar characteristics such as size and temperament traits which could in some cases be translated to similarity in costs.

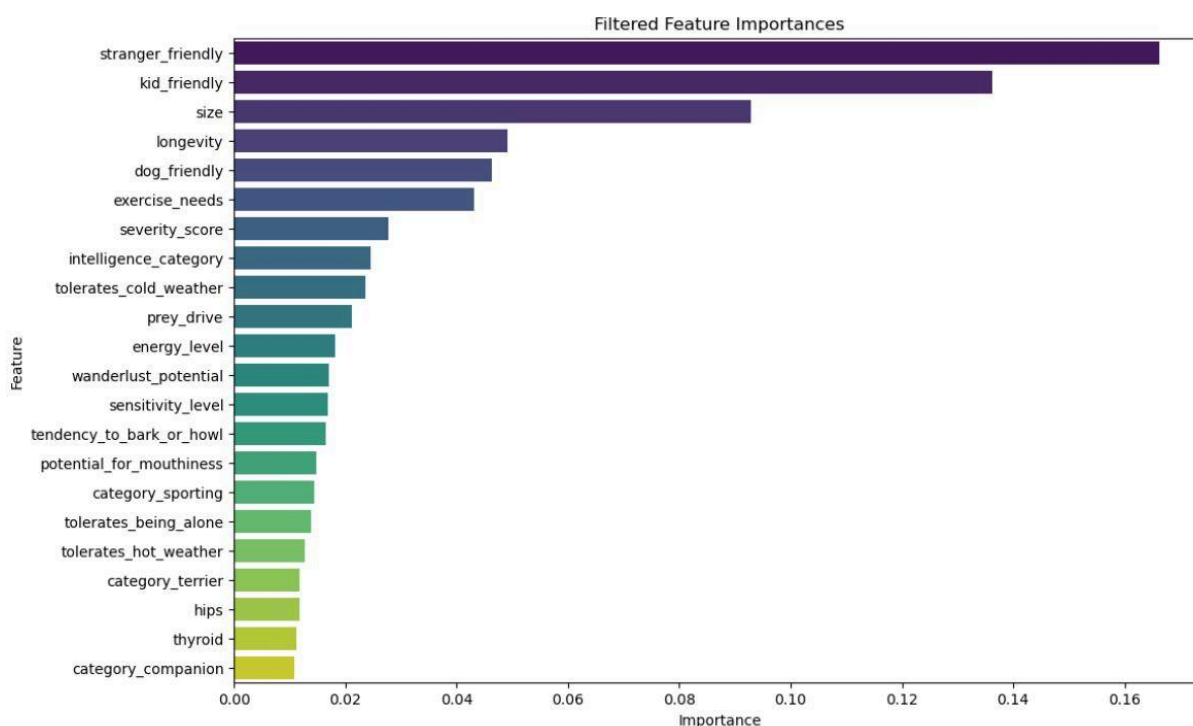
When it comes to physical traits such as 'grooming_required' and 'tolerates_cold_weather', there were implications of dogs with a high score translating to having a thick coat or long fur. The use of both variables would in such cases be unnecessary.

After reducing the feature set, the model still showed perfect scores of accuracy for the train and test sets:

New Model Performance (filtered features):

Training Accuracy: 1.0

Testing Accuracy: 1.0



Cross validation on filtered dataset (selected feature importance)

The cross-validation scores indicate that the model's performance is variable across different data samples, with an average performance of approximately 0.66. This result is indicative that the model may have still been overfitting, or that the features are not entirely representative of the target variable.

```
Cross-validation scores: [0.76377953 0.84839895 0.60488189 0.46094488
                          0.61503937]
Mean cross-validation score: 0.6586089238845144
```

Performance after Regularization

New Model Performance with Regularization (max_depth=10, min_samples_split=10, min_samples_leaf=5):

```
Training Accuracy: 0.9876771653543307
Testing Accuracy: 0.986745406824147
Precision: 0.9868037873845078
Recall: 0.986745406824147
F1-score: 0.9867435260246461
```

Cross-validation on model regularized

Training Classification Report:

	precision	recall	f1-score	support
High	1.00	0.98	0.99	37833
Low	0.98	0.99	0.99	38189
Lowest	0.99	1.00	1.00	38340
Medium	0.98	0.98	0.98	38038
accuracy			0.99	152400
macro avg	0.99	0.99	0.99	152400
weighted avg	0.99	0.99	0.99	152400

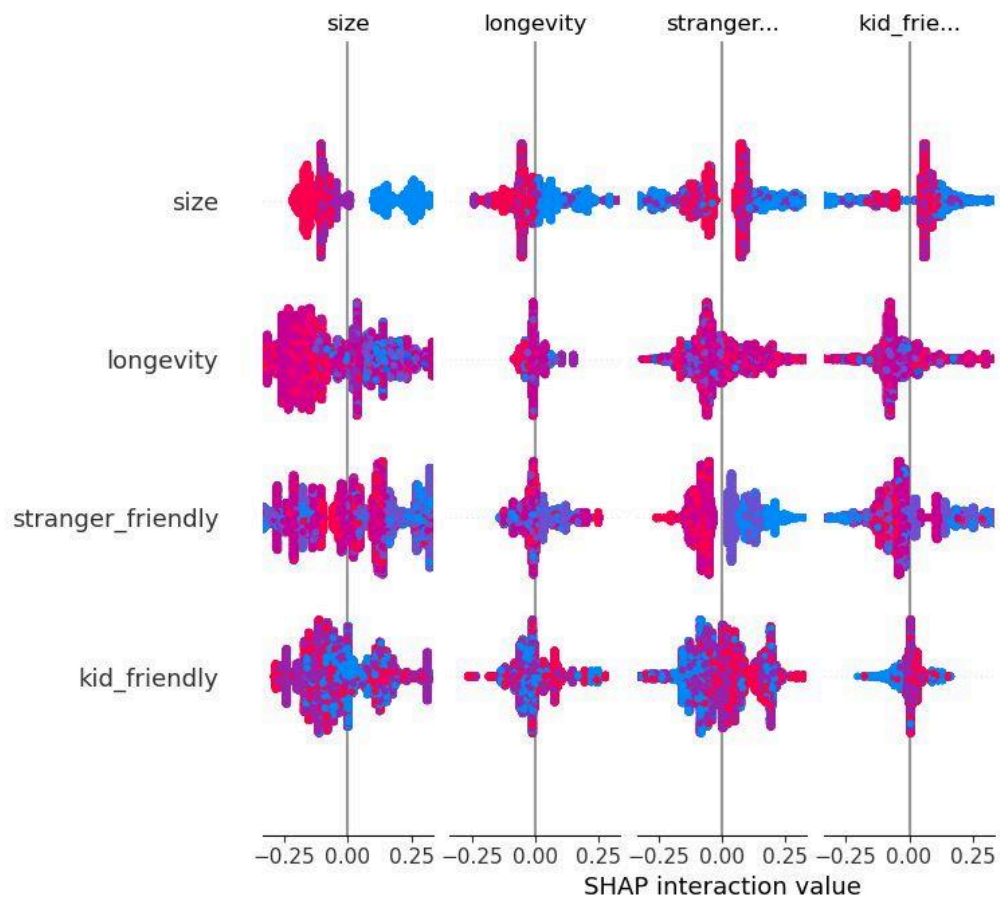
Testing Classification Report:

	precision	recall	f1-score	support
High	1.00	0.98	0.99	9339
Low	0.98	0.99	0.98	9726
Lowest	0.99	1.00	1.00	9416
Medium	0.98	0.98	0.98	9619
accuracy			0.99	38100
macro avg	0.99	0.99	0.99	38100
weighted avg	0.99	0.99	0.99	38100

```
Training ROC AUC: 0.9998189004379342
Testing ROC AUC: 0.999768424071771
```

The original model performance was showing potential overfitting, with metrics with 1.00 and few misclassifications in the confusion matrix. PCA did not significantly alter the model's performance, indicating the model was not sensitive to the correlation among features. After regularization, there was a slight reduction in metrics and a more realistic performance. High values close to 1.0 for both training and testing, indicating excellent discrimination ability (ROC AUC).

SHAP Chart (Random Forest)



From the SHAP chart, we can see that large size and poor longevity is positively correlated with a higher cost class, and that smaller size leads to better longevity and lower cost. For other variables, the interpretation was not so clear cut.

LIME (on Random Forest)

LIME stands for “local interpretable model-agnostic explanations”, and is a technique and generic framework to uncover black boxes and provides an explanation for predictions or recommendations with a local, interpretable model to explain each value. LIME values represent the extent to which each feature contributes to the prediction in a specific instance. Positive values indicate features with a higher probability of predicting the class (low, lowest, medium, high), while negative values indicate features with a low probability of predicting the target classes. Features with larger absolute LIME values (whether positive or negative) have a stronger impact on the model's predictive performance.

Below is the result of LIME explanations for the first four instances:

Instance 0

Predicted Class: **Lowest**

Actual Class: **Lowest**

Key Features Influencing Prediction:

- Stranger Friendly: -0.51
- Size: -1.19
- Sensitivity Level: -0.64
- Category Terrier: 2.18
- Dog Friendly: 0.33

The model's prediction for this instance is heavily influenced by the **stranger_friendly**, **size**, and **sensitivity_level** features. High values in **category_terrier** and **dog_friendly** also played a significant role.

Instance 1

Predicted Class: **Low**

Actual Class: **Low**

Key Features Influencing Prediction:

- Stranger Friendly: -2.49
- Kid Friendly: 1.17
- Size: -1.19
- Dog Friendly: 0.33
- Category Terrier: -0.46

The prediction was predominantly driven by the **stranger_friendly** and **kid_friendly** features. Negative values in **size** and **category_terrier**, along with a positive value in **dog_friendly**, were also influential.

Instance 2

Predicted Class: **High**

Actual Class: **High**

Key Features Influencing Prediction:

- Size: 1.14
- Stranger Friendly: -0.51
- Dog Friendly: -0.90
- Category Terrier: -0.46
- Sensitivity Level: 1.21

The **size** feature had the most significant impact, with positive contributions from **sensitivity_level** and negative contributions from **stranger_friendly**, **dog_friendly**, and **category_terrier**.

Instance 3

Predicted Class: **High**

Actual Class: **High**

Key Features Influencing Prediction:

- Size: 1.14
- Stranger Friendly: 0.48
- Sensitivity Level: 1.21
- Category Terrier: -0.46
- Kid Friendly: 1.17

The **size** feature again had a major positive impact, along with **sensitivity_level** and **kid_friendly**. Negative influence was noted from **category_terrier** while **stranger_friendly** contributed positively.

Feature Perturbation (sensitivity analysis)

A feature perturbation analysis was performed on the first 4 instances. This sensitivity analysis investigates the impact of perturbing individual features on the predictions of a classification model across four distinct instances. The perturbation was reached with a value of 1.0.

- **Common Sensitivities:** across all instances, features such as '**size**', '**intelligence_category**', '**sensitivity_level**', and certain behavioral traits (e.g. '**tendency_to_bark_or_howl**') consistently influenced prediction probabilities.
- **Impact Variation:** Perturbations in feature values often resulted in shifts towards adjacent prediction categories (e.g., from "Lowest" to "Low"). This indicates that the model's predictions are generally robust, with changes in feature values leading to incremental rather than drastic shifts in classification.
- **Complexity:** Some features showed minimal impact on predictions, suggesting that they either had a lower importance in the model's decision-making process or that their interactions with other features were not fully captured in isolation. For instance, features like '**category_companion**', '**category_sporting**', and '**category_working**' showed little to no change in prediction probabilities when perturbed.

We examined a few of the features more in detail:

- 'stranger_friendly' represents a dog's predisposition to friendliness (or alternatively, aggression) when encountering a stranger. Perturbation in the case of this feature had a noticeable impact across all instances, altering the prediction probabilities significantly. This suggests that this feature is quite influential in the model's decision-making process.
- The features 'size', 'severity_score', 'intelligence_category', and 'longevity' affect the prediction in most instances, which indicates their high relevance for the model.
- Features such as 'category_companion', 'category_sporting', and 'category_working' showed little to no change in prediction probabilities when perturbed across all instances. These features might therefore have minimal influence on the model's predictions or already be balanced within the model's considerations.

Decision Tree with Regularization

In order to understand the high performance of our Random Forest model, and with the idea of comparing it with a similar branching model, we decided to apply a Decision tree with regularization (max_depth=10, min_samples_split=10, min_samples_leaf=5). Below is an analysis of the Decision Tree performance, along with a classification report.

Decision Tree Model Performance:

```
Training Accuracy: 0.9662073490813649
Testing Accuracy: 0.9642782152230971
Precision: 0.9654977356639471
Recall: 0.9642782152230971
F1-score: 0.9644424253729907
```

```
Cross-validation accuracy scores: [0.72440945 0.90981627 0.83685039
0.52007874 0.6244357 ]
Mean cross-validation accuracy: 0.7231181102362205
```

Training Classification Report (Decision Tree):

	precision	recall	f1-score	support
High	0.99	0.95	0.97	37833
Low	0.97	0.94	0.96	38189
Lowest	1.00	0.99	0.99	38340
Medium	0.91	0.98	0.94	38038
accuracy			0.97	152400
macro avg	0.97	0.97	0.97	152400
weighted avg	0.97	0.97	0.97	152400

```

Testing Classification Report (Decision Tree):
              precision    recall  f1-score   support

   High         0.99         0.94         0.96         9339
    Low         0.97         0.94         0.96         9726
  Lowest         0.99         0.99         0.99         9416
   Medium        0.91         0.98         0.94         9619

 accuracy              0.96         38100
 macro avg         0.97         0.96         0.96         38100
weighted avg         0.97         0.96         0.96         38100

```

```

Training ROC AUC (Decision Tree): 0.998118718690002
Testing ROC AUC (Decision Tree): 0.9978297076792021

```

Hyperparameter Tuning with RandomizedSearchCV (second-order regularization):

```

Random Forest Best Parameters: {'n_estimators': 50, 'min_samples_split':
20, 'min_samples_leaf': 10, 'max_depth': 20}
Random Forest Best Cross-validation Score: 0.9955774278215224
Decision Tree Best Parameters: {'min_samples_split': 20,
'min_samples_leaf': 2, 'max_depth': 20}
Decision Tree Best Cross-validation Score: 0.9992979002624672

```

The Decision Tree model shows a higher best cross-validation score compared to the Random Forest model.

The **Decision Tree model**, after hyperparameter tuning, shows a very high cross-validation score, indicating it performs exceptionally well in terms of fitting the training data. The ROC AUC score on the test set is very close to 1, suggesting that it has a very high capability in distinguishing between the classes. The extremely high cross-validation score could indicate that the model is overfitting to the training data. It's crucial to verify that this high performance generalizes well to new, unseen data, which is somewhat confirmed by the testing metrics.

The **Random Forest model** also performs well, with a high cross-validation score and similar testing metrics to the Decision Tree model. However, it has slightly lower cross-validation performance compared to the Decision Tree. It shows slightly lower cross-validation scores but maintains a robust performance on the test set. This indicates that it might be better at generalizing compared to the Decision Tree model, as ensemble methods often provide better generalization.

Random Forest Performance

- The Random Forest model showed high performance in terms of accuracy, precision, recall, F1-score, and ROC AUC, both on the training and testing datasets.

Decision Tree Performance

- The Decision Tree model has slightly lower overall performance metrics compared to the Random Forest.

Comparative Analysis

Accuracy:

Random Forest shows higher accuracy on both training (98.8%) and testing (98.7%) compared to the Decision Tree model (96.6% training, 96.4% testing).

Precision, Recall, F1-Score:

Random Forest shows higher precision, recall, and F1-scores across all classes compared to the Decision Tree model, reflecting improved overall performance.

ROC AUC:

Random Forest has slightly higher ROC AUC scores on both training (0.9998) and testing (0.9998) compared to the Decision Tree model (0.998 training, 0.998 testing). This indicates better discrimination between classes.

Classification Reports:

Random Forest shows a more balanced and consistently high performance across all classes. The Decision Tree model shows slightly lower precision and recall for the 'Medium' class, while the Random Forest model handles all classes with near-perfect scores.

Cross-Validation Accuracy:

The Decision Tree model's cross-validation scores indicate more variability, while the Random Forest model's performance is stable and high, suggesting better generalization.

Random Forest for Regression

The Random Forest regression model provides insights into how well the features predict the continuous target variable. The idea behind the implementation of both, regression and classification using random forest, is based on the possibility of understanding the underlying patterns. The Random Forest Regressor is well-suited for predicting continuous

outcomes by averaging the predictions of multiple decision trees, which reduces overfitting and improves generalization. On the other hand, the Random Forest Classifier, applied to the discretized version of the target variable, can categorize data into predefined classes, enabling a different perspective on the relationships within the data. This approach provides insights into whether the continuous target's granularity significantly affects predictive capabilities.

Random Forest before dimension reduction

Model Performance Before PCA

Train RMSE: 0.9759639474081109

Test RMSE: 1.220350026161229

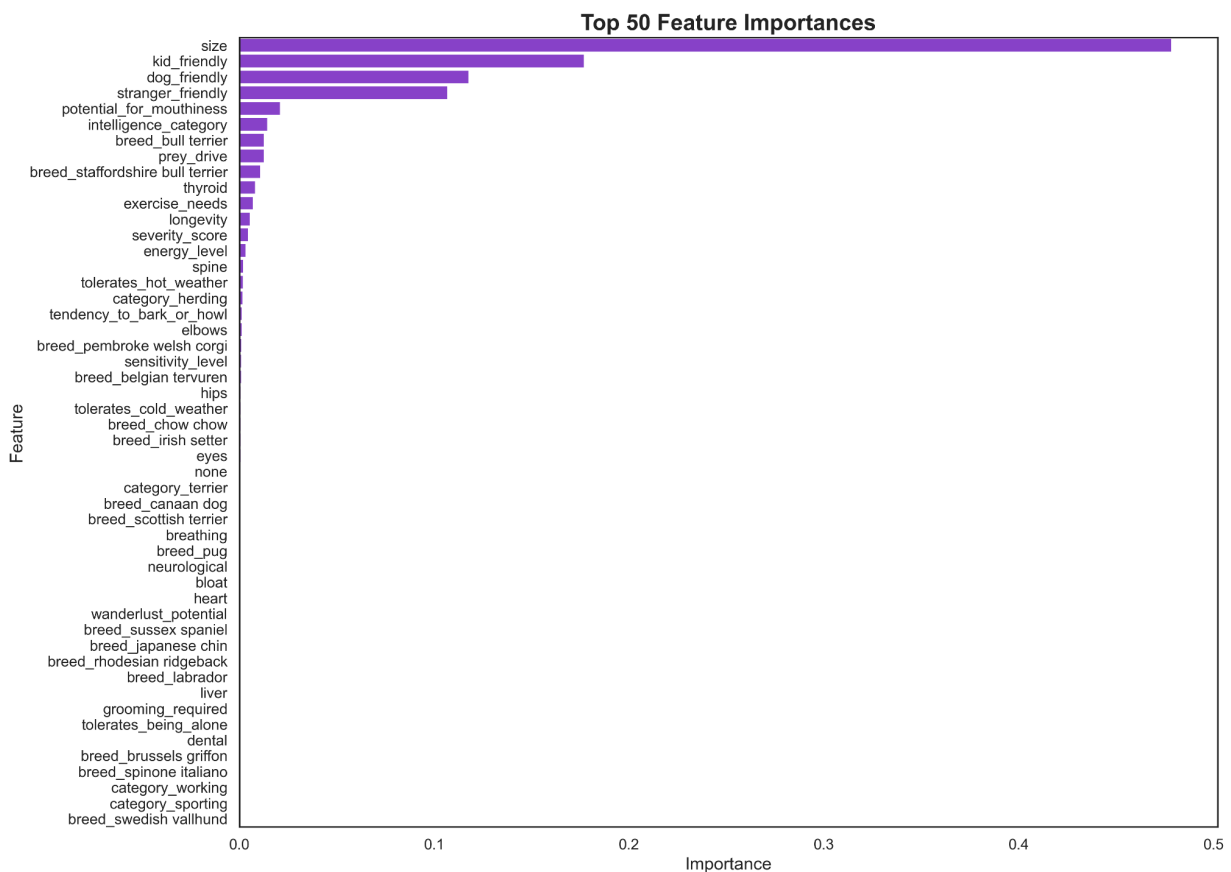
Train MAE: 0.02736423884514441

Test MAE: 0.0490501312335959

Train R2: 0.9999979958977231

Test R2: 0.9999968016225496

Feature performance before dimension reduction (fig.1)



Random Forest after dimension reduction

Model Performance After PCA

Train RMSE: 2.692884150235194

Test RMSE: 6.6682551698700525

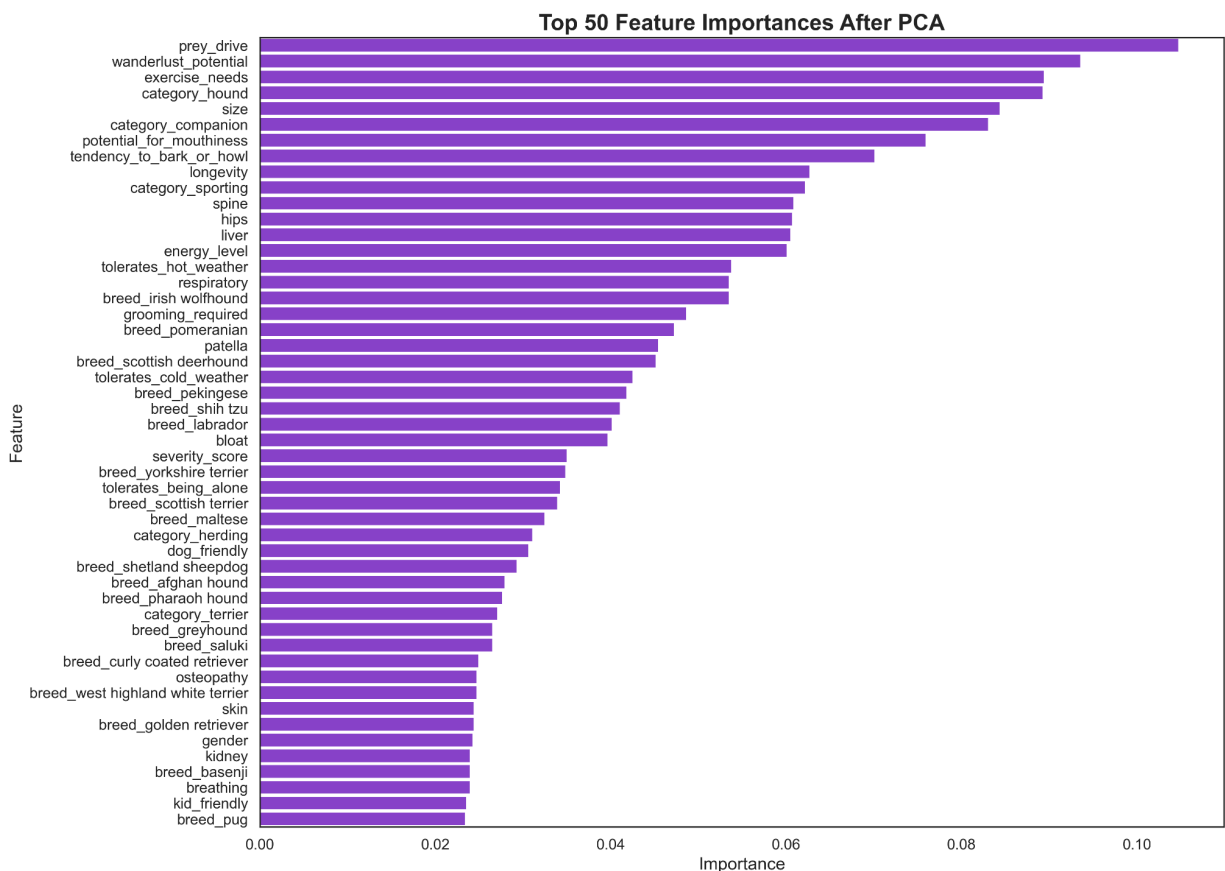
Train MAE: 0.1446847112860892

Test MAE: 0.38157244094488185

Train R2: 0.9999847423491678

Test R2: 0.999904503972491

Feature performance after dimension reduction (fig.2)



There is a performance drop when PCA is used. This is common if the principal components do not capture all relevant aspects of the data. Feature importance can shift significantly due to the transformation of features into principal components, reflecting the new representation of features rather than their original individual importance (fig. 2)

We checked different variance thresholds for PCA to see if they impact the model performance:

Loaded Metrics for PCA with 90% Variance Threshold

Train RMSE: 2.7318677650216348
Test RMSE: 6.80308857351627
Train MAE: 0.14684133858267714
Test MAE: 0.39470629921259853
Train R2: 0.9999842973959822
Test R2: 0.9999006030322173

Top Features After PCA:

	Feature	Importance
20	prey_drive	0.105095
22	wanderlust_potential	0.093981
23	exercise_needs	0.089923
6	category_hound	0.089609
4	category_companion	0.083959
3	size	0.083733
19	potential_for_mouthiness	0.075135
21	tendency_to_bark_or_howl	0.069945
2	longevity	0.061893
7	category_sporting	0.061769

Loaded Metrics for PCA with 85% Variance Threshold

Train RMSE: 2.690077005942443
Test RMSE: 6.71826439396304
Train MAE: 0.14441496062992132
Test MAE: 0.3884493438320211
Train R2: 0.9999847741426658
Test R2: 0.9999030662380424

Top Features After PCA:

	Feature	Importance
20	prey_drive	0.105780
22	wanderlust_potential	0.094860
23	exercise_needs	0.090286
6	category_hound	0.090184
4	category_companion	0.084826
3	size	0.084719
19	potential_for_mouthiness	0.075174
21	tendency_to_bark_or_howl	0.069339
2	longevity	0.063106
7	category_sporting	0.061818

Loaded Metrics for PCA with 80% Variance Threshold

```
Train RMSE: 2.6667837947127966
Test RMSE: 7.233013082524359
Train MAE: 0.14387696850393705
Test MAE: 0.40267506561679794
Train R2: 0.9999850366805967
Test R2: 0.9998876431935809
```

Top Features After PCA:

	Feature	Importance
20	prey_drive	0.108185
6	category_hound	0.098411
22	wanderlust_potential	0.098324
23	exercise_needs	0.089101
4	category_companion	0.086506
3	size	0.084547
19	potential_for_mouthiness	0.081273
21	tendency_to_bark_or_howl	0.074320
34	hips	0.065222
44	spine	0.064348

Applying PCA has changed the importance of features in a way that affects the model's performance and the interpretation of feature contributions. Before PCA, the most important features were 'size', 'kid_friendly', 'dog_friendly', 'stranger_friendly', 'potential_for_mouthiness', and 'intelligence'.

After PCA, the top features contributing to the principal components are 'prey_drive', 'category_hound', 'wanderlust_potential', 'exercise_needs', 'category_companion', 'size, potential_for_mouthiness', 'tendency_to_bark_or_howl, hips', and 'spine'.

Binarizing the Data

To analyze the effects of discretizing the continuous target variable into a binary format, we utilized the *dogs_expenses.csv* dataset. This involved calculating the 'health_severity_score' from health insurance costs and severity scores, subsequently binarizing it based on the median, categorizing breeds into low (0) and high (1) severity groups. This strategy aimed to reveal how well models perform under binary classification compared to continuous targets.

```

### Feature Importances ###
              feature importance
0              longevity  0.098617
2              size_category  0.097173
6      tolerates_cold_weather  0.076247
3      intelligence_category  0.067648
14             exercise_needs  0.060961
13      wanderlust_potential  0.056421
10    potential_for_mouthiness  0.053552
11              prey_drive  0.053138
7      tolerates_hot_weather  0.048574
15             energy_level  0.044288
12    tendency_to_bark_or_howl  0.043209
5      tolerates_being_alone  0.041560
4      sensitivity_level  0.036072
9      friendly_towardstrangers  0.030588
1      grooming_required  0.030143
23      category_Working Dogs  0.028650
17              age  0.027278
22      category_Terrier Dogs  0.022014
8    incredibly_kifriendly_dogs  0.021601
19      category_Herding Dogs  0.017662
21      category_Sporting Dogs  0.014496
20      category_Hound Dogs  0.011657
18      category_Companion Dogs  0.011102
16              gender  0.007348

```

From the feature importance rankings, it is evident that variables such as 'longevity,' 'size_category,' and 'tolerates_cold_weather' are significant predictors, indicating their strong influence on health severity among dog breeds.

Classification Reports

Listed below are a collection of classification reports from various techniques:

```

### Classification Report-Random Forest###
              precision    recall  f1-score   support

         0               1.00      1.00      1.00      20419
         1               1.00      1.00      1.00      17681

 accuracy                   1.00      38100
 macro avg               1.00      1.00      1.00      38100
weighted avg               1.00      1.00      1.00      38100

```

ROC AUC Score: 1.0000

```

### Cross-Validation Scores (AUC)- Random forest###

```

```

[0.81284402 0.79371676 0.74240547 0.67848494 0.69157041]
Mean AUC: 0.7438

```

Standard Deviation AUC: 0.0534
Model and scaler saved successfully.

```
### Classification Report - Logistic regression ###
              precision    recall  f1-score   support

     0           1.00        1.00        1.00        20419
     1           1.00        1.00        1.00        17681

 accuracy              1.00              38100
 macro avg           1.00        1.00        1.00        38100
 weighted avg        1.00        1.00        1.00        38100
```

ROC AUC Score: 1.0000

```
### Cross-Validation Scores (AUC)- Logistic regression ###

[0.64048978 0.72754026 0.63176795 0.54849899 0.52785082]
Mean AUC: 0.6152
Standard Deviation AUC: 0.0716
Model and scaler saved successfully.
```

```
### Classification Report - LR with class weights ###
              precision    recall  f1-score   support

     0           0.72        0.75        0.74        20419
     1           0.70        0.66        0.68        17681

 accuracy              0.71              38100
 macro avg           0.71        0.71        0.71        38100
 weighted avg        0.71        0.71        0.71        38100
```

ROC AUC Score: 0.8069

```
### Cross-Validation Scores (AUC)- LR with Balanced Class Weights ###

[0.63948701 0.73019646 0.63120315 0.55111711 0.52770767]
Mean AUC: 0.6159
Standard Deviation AUC: 0.0719
Balanced model and scaler saved successfully.
```

```
### Classification Report for Best Logistic Regression (Grid search CV)
###
              precision    recall  f1-score   support

     0           0.73        0.69        0.71        20419
     1           0.66        0.71        0.68        17681

 accuracy              0.70              38100
 macro avg           0.70        0.70        0.70        38100
 weighted avg        0.70        0.70        0.70        38100
```

ROC AUC Score: 0.8045
 Best Parameters: {'C': 0.001, 'penalty': 'l2', 'solver': 'liblinear'}
 Best Cross-Validation AUC: 0.6229830458624127

Cross-Validation Scores (AUC)- **Gradient Boosting Classifier** ###
 [0.70585514 0.75835648 0.68035704 0.66656849 0.70124045]
 Mean AUC: 0.7025
 Standard Deviation AUC: 0.0314

Best Parameters: {'n_estimators': 200, 'max_depth': 7, 'learning_rate': 0.01}
 Best Cross-Validation AUC- **Randomized search CV:** 1.0
 Best GBM model and scaler saved successfully.
 Training Accuracy: 0.9953608923884515
 Testing Accuracy: 0.9949606299212599

Classification Report for **Best GBM**

	precision	recall	f1-score	support
0	1.00	0.99	1.00	20419
1	0.99	1.00	0.99	17681
accuracy			0.99	38100
macro avg	0.99	1.00	0.99	38100
weighted avg	1.00	0.99	0.99	38100

ROC AUC Score: 1.0000

GradientBoostingClassifier(learning_rate=0.01, max_depth=7,
 n_estimators=200,random_state=42)

Training Classification Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	81581
1	1.00	1.00	1.00	70819
accuracy			1.00	152400
macro avg	1.00	1.00	1.00	152400
weighted avg	1.00	1.00	1.00	152400

Testing Classification Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	20419
1	1.00	1.00	1.00	17681
accuracy			1.00	38100
macro avg	1.00	1.00	1.00	38100
weighted avg	1.00	1.00	1.00	38100

Training ROC AUC: 1.0
 Testing ROC AUC: 1.0

Classification Model Performance

Random Forest Classifier:

- Achieved perfect precision, recall, and F1 scores of 1.00 for both classes, indicating flawless performance on the training data.
- Cross-validation AUC scores varied significantly, with a mean AUC of 0.7438 and a standard deviation of 0.0534, suggesting some variability in model performance on different data splits.

Logistic Regression:

- Initial logistic regression also showed perfect scores on training data but exhibited a lower mean AUC of 0.6152 with higher variability (standard deviation of 0.0716).
- Adjusting for class weights improved the AUC to 0.8069, reflecting better handling of class imbalance.

Gradient Boosting Classifier (GBM):

- Demonstrated robust performance with a mean AUC of 0.7025 and a relatively low standard deviation of 0.0314.
- The GBM model also achieved perfect ROC AUC scores on both training and testing datasets, showcasing its ability to capture complex relationships even with binary targets.

The **Random Forest Classifier's** perfect scores might indicate overfitting, while its cross-validation results suggest moderate performance variability. **Logistic Regression** showed limitations in handling class imbalance, which was partially mitigated by adjusting class weights.

The **Gradient Boosting Classifier** emerged as a strong performer, maintaining high accuracy and stability across different validation sets. While the Gradient Boosting Classifier is performing exceptionally well, it is crucial to validate and fine-tune the model further to ensure its robustness and generalizability.

K-Nearest Neighbors

K-Nearest Neighbours is another algorithm that was employed on the dataset. It was used on *clean_split_data.csv* (created by splitting *combined_data.csv*) with encoded breeds and *nobreed_data.csv* - without breeds.

Splitting the Dataset

A third strategy was employed in which we split the dataset vertically, to include only behaviour-related features, and alternatively only health-related features. The aim here was to determine whether certain types of features were having an exaggerated influence on the model's accuracy. We also decided to employ additional calculations for training costs based on certain behavioural traits and had previously used 'severity_score' to represent 'genetic_ailments' following their one hot encoding, and no financial multiplier for behaviour-related variables such as 'sensitivity_level', 'kid_friendly' etc.

Behaviour Only (with Breed)

Behaviour costs added to the 'final_yearly_cost' were calculated using regular training costs positively correlated with a high score for most behaviour variables. A negatively correlated score for 'kid_friendly', 'stranger_friendly', 'dog_friendly' was calculated using an increased multiplication of the regular training costs. As we did not have raw, uncalculated data, it was likely the reason for the model performing too perfectly.

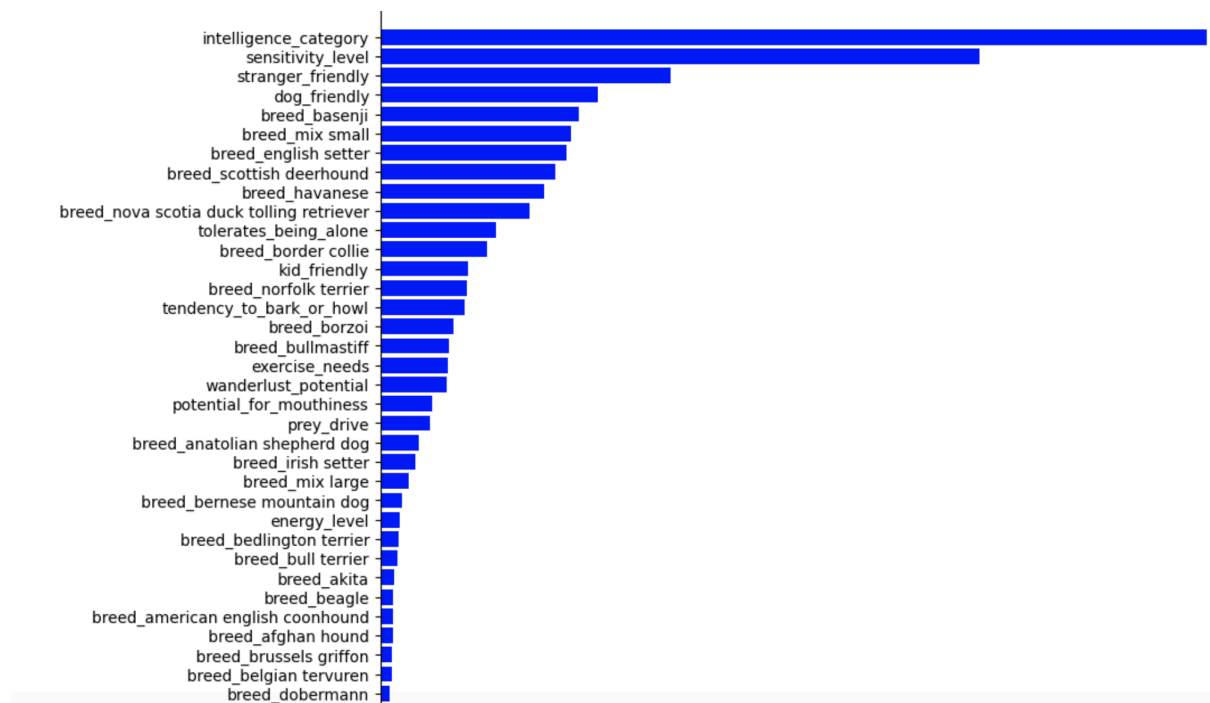
```
BEHAVIOUR DATASET
Accuracy Score: 0.9997900262467192
Classification Report:
              precision    recall  f1-score   support

         2           1.00      1.00      1.00        3366
         3           1.00      1.00      1.00        3275
         4           1.00      1.00      1.00       31459

 accuracy                   1.00      38100
 macro avg           1.00      1.00      1.00      38100
weighted avg           1.00      1.00      1.00      38100

Confusion Matrix:
[[ 3360     6     0]
 [     0  3274     1]
 [     0     1 31458]]
Model Accuracy: 99.98%
```

LightGBM (with Breed) Behaviour Feature Importance Ranking



Behaviour Only (without Breed)

Removing the individual breeds did not improve the model's performance, likely due to the overall low importance of breeds and the cost figure calculation being simplistic as we lacked raw data.

BEHAVIOUR DATASET

Accuracy Score: 1.0

Classification Report:

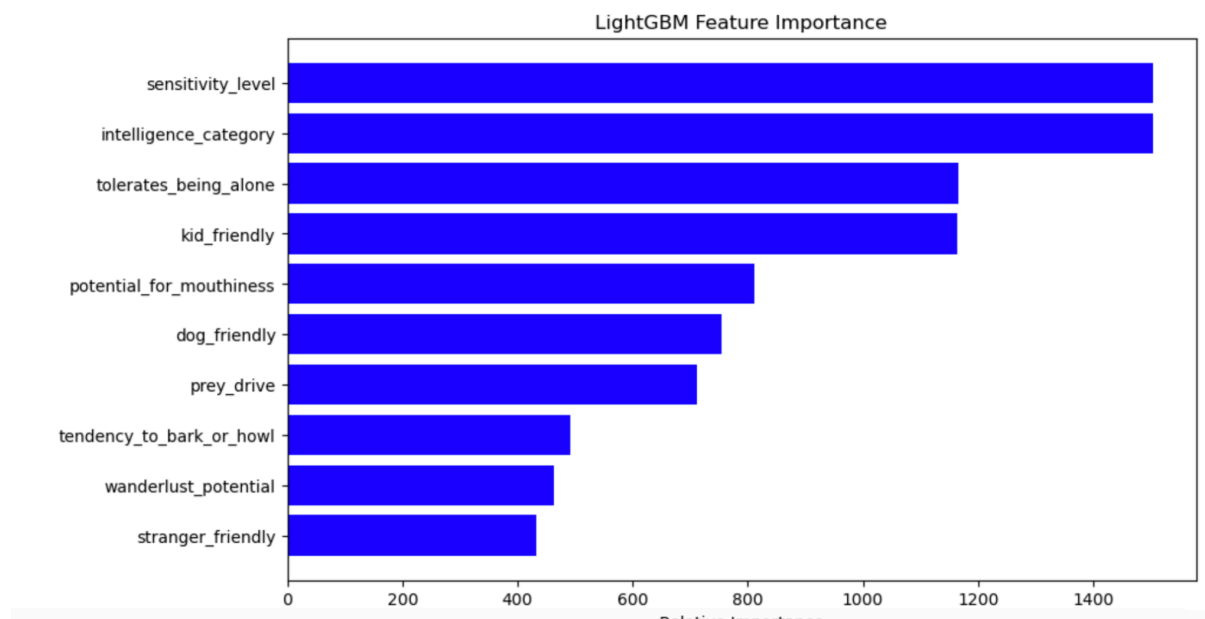
	precision	recall	f1-score	support
2	1.00	1.00	1.00	192
3	1.00	1.00	1.00	1145
4	1.00	1.00	1.00	36763
accuracy			1.00	38100
macro avg	1.00	1.00	1.00	38100
weighted avg	1.00	1.00	1.00	38100

Confusion Matrix:

```
[[ 192    0    0]
 [   0 1145    0]
 [   0    0 36763]]
```

Model Accuracy: 100.00%

LightGBM (without Breed) Behaviour Feature Importance Ranking



Ailments Only (with Breed)

The ailments only half of the dataset performed comparatively well. This could be due to the complexity of the different calculations related to health data.

AILMENTS DATASET

Accuracy Score: 0.9896850393700788

Classification Report:

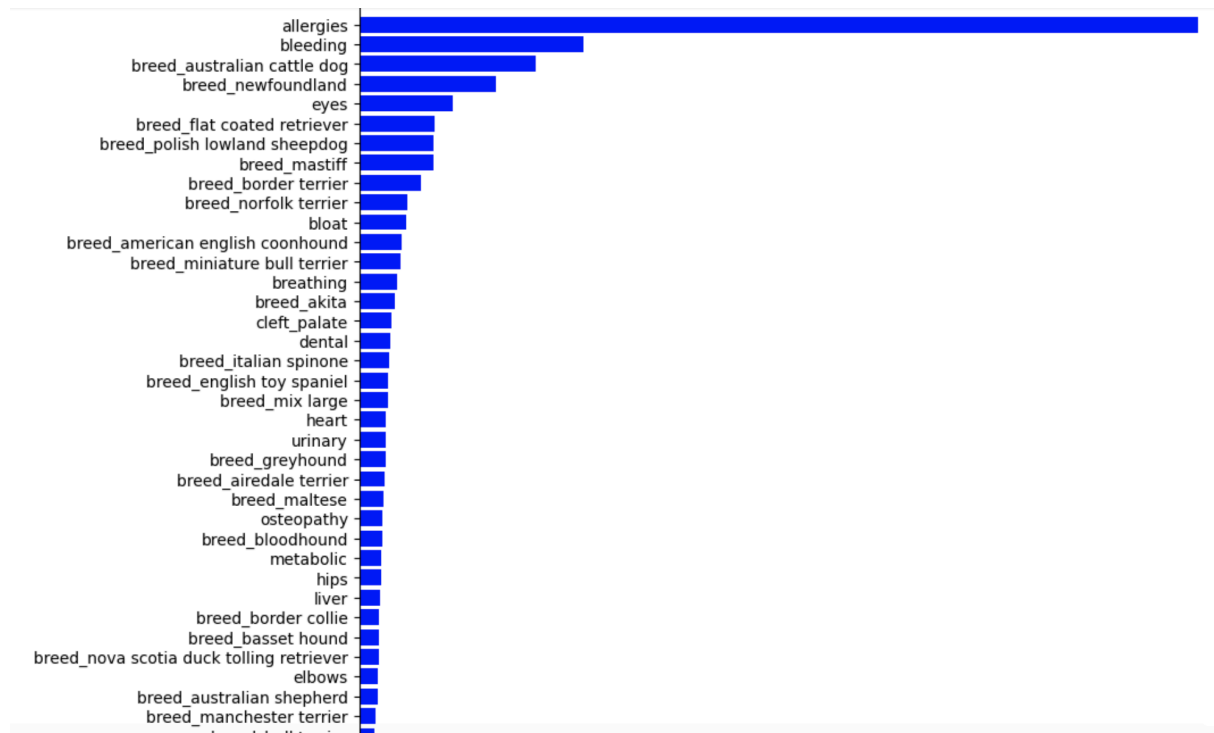
	precision	recall	f1-score	support
2	0.00	0.00	0.00	192
3	1.00	0.82	0.90	1145
4	0.99	1.00	0.99	36763
accuracy			0.99	38100
macro avg	0.66	0.61	0.63	38100
weighted avg	0.98	0.99	0.99	38100

Confusion Matrix:

```
[[ 0  0 192]
 [ 0 944 201]
 [ 0  0 36763]]
```

Model Accuracy: 98.97%

LightGBM (with Breed) Ailments Feature Importance Ranking



Ailments Only (without Breed)

Removing breeds appeared to reduce the overfitting of the model.

AILMENTS DATASET

Accuracy Score: 0.8589501312335958

Classification Report:

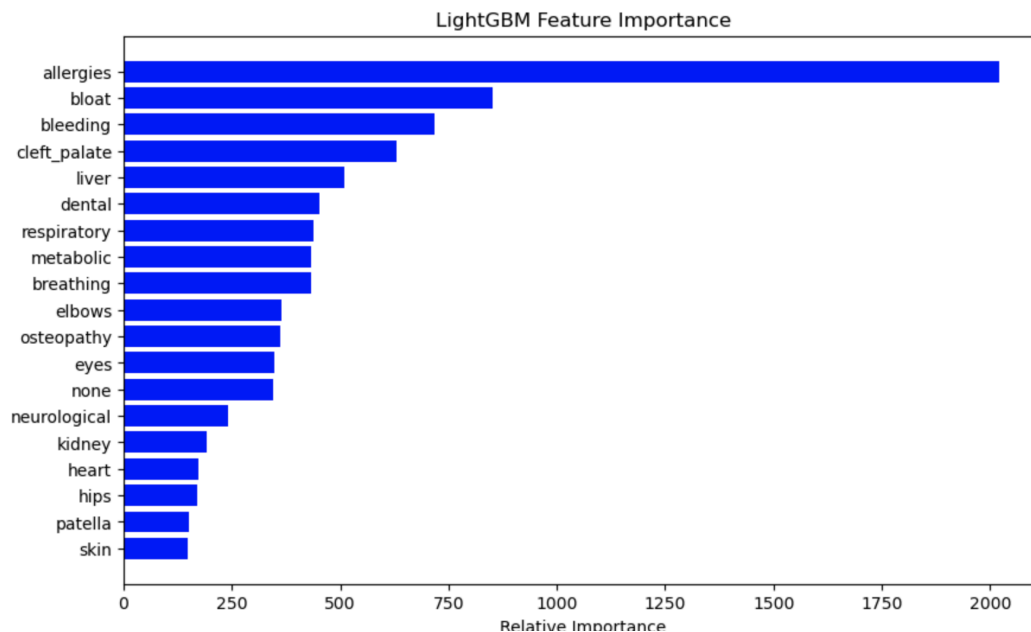
	precision	recall	f1-score	support
2	0.61	0.71	0.65	3366
3	0.49	0.13	0.21	3275
4	0.90	0.95	0.92	31459
accuracy			0.86	38100
macro avg	0.67	0.60	0.60	38100
weighted avg	0.84	0.86	0.84	38100

Confusion Matrix:

```
[[ 2393   74  899]
 [  364  437 2474]
 [ 1187  376 29896]]
```

Model Accuracy: 85.90%

LightGBM (without Breed) Ailments Feature Importance Ranking



Support Vector Machine

Change of Target Variable

In an effort to understand the interrelation of different variables and their influence on 'breed', a final strategy was employed in which the target variable was changed to 'breed' and the SVM algorithm was used. The *clean_split_data.csv* dataset was used, assigning breeds numeric labels instead of one hot encoding. Two approaches were taken: feature importance ranking based on a specific breed and feature importance ranking common to all breeds.

The feature ranking for an individual breed is pictured below:

Feature ranking:

1. Feature 'tendency_to_bark_or_howl' (Weight: 0.06265928754222089)
2. Feature 'grooming_required' (Weight: 0.05698572439392033)
3. Feature 'category_companion' (Weight: 0.055208839980648204)
4. Feature 'intelligence_category' (Weight: 0.04863402542765306)
5. Feature 'hips' (Weight: 0.04390942863089128)
6. Feature 'age' (Weight: 0.00013784036702132324)
7. Feature 'gender' (Weight: 4.6331362080209715e-09)
8. Feature 'potential_for_mouthiness' (Weight: 3.469446951953614e-18)
9. Feature 'osteopathy' (Weight: 0.0)
10. Feature 'longevity' (Weight: 0.0)

The confusion matrix of all breed feature importance follows, along with the classification report. In both cases, the classification reports clearly indicate overfitting, leading us to the conclusion that breed has a low importance in model prediction in relation to the explanatory features.

```

Confusion Matrix:
[[ 300    0    0    0    0    0    0    0    0    0    0]
 [    0 3917    0    0    0    0    0    0    0    0    0]
 [    0    0 5663    0    0    0    0    0    0    0    0]
 [    0    0    0 7655    0    0    0    0    0    0    0]
 [    0    0    0    0 257    0    0    0    0    0    0]
 [    0    0    0    0    0 3579    0    0    0    0    0]
 [    0    0    0    0    0    0 2054    0    0    0    0]
 [    0    0    0    0    0    0    0 3544    0    0    0]
 [    0    0    0    0    0    0    0    0 4180    0    0]
 [    0    0    0    0    0    0    0    0    0 4499    0]
 [    0    0    0    0    0    0    0    0    0    0 2452]]

Model Accuracy: 100.00%
Top feature ranking (common to predicting all breeds):
1. Feature 'wanderlust_potential' (Importance: 13.442221909705687)
2. Feature 'thyroid' (Importance: 12.662010224283588)
3. Feature 'prey_drive' (Importance: 12.491465372726259)
4. Feature 'intelligence_category' (Importance: 11.65193807242324)
5. Feature 'potential_for_mouthiness' (Importance: 11.636687675919866)
6. Feature 'tendency_to_bark_or_howl' (Importance: 11.468628551244182)
7. Feature 'dog_friendly' (Importance: 11.12026478100518)
8. Feature 'tolerates_being_alone' (Importance: 10.862369566127839)
9. Feature 'size' (Importance: 10.735131079813433)
10. Feature 'tolerates_cold_weather' (Importance: 10.242371969039608)
11. Feature 'sensitivity_level' (Importance: 9.41790420423412)
12. Feature 'longevity' (Importance: 9.393652070283979)
13. Feature 'energy_level' (Importance: 9.022004070539282)
14. Feature 'exercise_needs' (Importance: 9.001805284613237)
15. Feature 'hips' (Importance: 8.433037636016229)
16. Feature 'none' (Importance: 8.192494553855239)
17. Feature 'eyes' (Importance: 8.140009618399656)
18. Feature 'grooming_required' (Importance: 7.974561737652105)
19. Feature 'heart' (Importance: 7.696262849803286)
20. Feature 'spine' (Importance: 7.627799903815624)

```

Key Takeaways

Since ‘**breed**’ was proven to have low importance in model predictions regardless of algorithm and model performance boosting techniques, the switch to ‘**breed_category**’ was made. What remains to be seen is whether further variables should be dropped in the continued tuning of our model and the subsequent effect on our business case. Our original problem space was predicting cost categories on the basis of individual breeds, but dog owners are unlikely to be able to choose the correct breed category when using our app. A further point to highlight is mixed breeds, which are further disadvantaged in the case of breed groups as they simply belong to “other” yet make up a significant portion of the dog population.

Conclusion

From the beginning to the end of the project, we were faced with various challenges. We had to make fundamental decisions related to data collection and building a workable dataset, which continued into extensive feature engineering and finally, a reorientation of the business case. Due to the innovative nature of the technical product for which the project was developed, it was not possible to find “ready-made” datasets to work with. The data we were able to obtain was skewed from the financial perspective towards the US and UK, which are not comparable to Germany due to being much more mature pet industry markets. When it came to health data, the majority of open data sources were related to longevity or chronic illness use cases, and so there was overrepresentation of certain breeds, which needed to be balanced. And although behaviour data was available, there was no data which linked it to dog ownership expenses and thus we had to develop this link ourselves through calculations and primary research. Feature engineering thus took longer than expected.

We were advised by our mentor to try a minimum of four different machine learning models and to go from there. However, we encountered a common problem among all algorithms: overfitting. At times it seemed as though we were blindly trying out various techniques in order to overcome this challenge, but perhaps we should have invested time in understanding how to better prepare the data for specific models and use cases. This was a skill not touched on within the bootcamp - understandably, given that the majority of students chose a standard project which came with a ready dataset, whereas we had to build our own.

As the project progressed, we tried to play to our strengths with the task division. Due to her domain expertise, Natasha focused on feature engineering and variable selection, creating various dataset versions. Veronica as the technical lead performed the advanced model tuning, going beyond classification to include regression, utilizing the different datasets. A further frustrating challenge we encountered was limitation by computational power and thus there were limits to how complex our models could be. Code optimization and best practices were unfortunately not touched upon in our course materials.

Our objectives were mostly met in that we were able to determine the key criteria correlated with dog ownership expenses. However, there are limitations to detecting risk factors in individual dogs as we focused on purebreds and yet a large percentage of the dog owner population have mongrels. We considered removing breed altogether and changing to breed categories, however, this does not solve the problem. It is also unlikely that dog owners would be able to objectively quantify their dog's traits as they are presented in the dataset. If the model is to be integrated with the platform, then further feature engineering must take place in order to transform traits into binaries understandable by the average layperson. Further considerations include a new use case targeting potential savings, rather than alerting to risk. This could take the form of promoting certain decisions such as buying insurance - comparing the average health-related expenses with and without it. Lastly, primary research and text mining on social media could be used to further enrich the dataset and remove bias towards health.

Bibliography

Literature

- [Digital Pet Care Industry: European Market, focus on Germany, France & Bulgaria \(Paw à Peau, 2022\)](#)
- [Increasing adoption rates at animal shelters: a two-phase approach to predict length of stay and optimal shelter allocation \(Bradley & Rajendran, 2021\)](#)
- [Better animal welfare through insurance \(Zenz-Spitzweg, 2021\)](#)
- [The Economic Significance of Pet Ownership in Germany \(Ohr, 2019\)](#)
- [The Economic Significance of Pet Ownership in Germany \(Ohr, 2014\)](#)

Primary Datasets

<https://www.kaggle.com/datasets/paultimothymooney/best-in-show-data-about-dogs>
(foundation)
<https://www.kaggle.com/datasets/jainaru/dog-breeds-ranking-best-to-worst>
<https://www.kaggle.com/datasets/iqamharisfahromi/pet-sales>
<https://www.kaggle.com/datasets/jahangirraina/pet-food-customer-orders-online>
https://data.cityofnewyork.us/Health/DOHMH-Dog-Bite-Data/rsgh-akpg/about_data
<https://www.kaggle.com/datasets/rtatman/animal-bites>
https://www.kaggle.com/datasets/agarwalyashhh/dog-adaptability?select=training_data.csv
<https://www.kaggle.com/datasets/jackdaoud/animal-shelter-analytics>
https://data.bloomington.in.gov/Public-Works/Animal-Shelter-Animals/e245-r9ub/about_data

Supplementary Datasets

(used for enrichment, along with the German pet ownership studies)

<https://rvc-repository.worktribe.com/output/1375552>
<https://data.world/the-pudding/adoptable-dogs-on-petfinder-in-the-us/workspace/file?filename=alIDogDescriptions.csv>
<https://rvc-repository.worktribe.com/output/1375689>
<https://rvc-repository.worktribe.com/output/1558210>

 Pet health pricing model

Appendix

[Miro](#) - Dataset schema and project milestones

[Data enrichment](#) - notes, figures sourced from German studies

[Data audit](#) - analysis of variables

Github Code File	Description	Purpose
Dog_expenses_clean.ipynb	Data Cleaning I	Prepare data for exploration
pet_expense_visualization.py	Foundation dataset exploration	Visualizing data relationships
cleaned_df_enriched.ipynb	Data Cleaning II	Prepare data for modelling, first enrichment
Ailment_financials.ipynb	Feature engineering I	Add cost calculations for individual ailments
add_cost_encode_ailments.py	Feature engineering II	Encode ailments, refine cost calculation
clean_ailments_training.py	Feature engineering III	Add training-related costs
modelling_svm.py	SVM (whole dataset)	Testing SVM as a model
Knn_split_behaviour.ipynb	KNN (behaviour variables only, with and without breeds)	Test only behaviour on predicting cost classification
Knn_split_ailments.ipynb	KNN (ailment variables only, with and without breeds)	Test only ailments on predicting cost classification
SVC_target_breeds.ipynb	SVM (predicting breeds)	Test different target variable (breeds)
Recategorizing_breeds.ipynb	Recategorizing breeds	Feature engineering (final)
Classification_binary_target.ipynb	Random forest (classification)	Testing binary classification
Classification_final.ipynb	Final chosen model (classification)	Advanced tuning
Regression_final.ipynb	Final chosen model (regression)	Advanced tuning