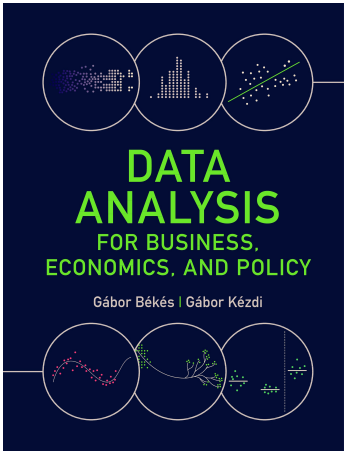
14. Prediction process

Gábor Békés

Data Analysis 3: Prediction

2020

Slideshow for the Békés-Kézdi Data Analysis textbook

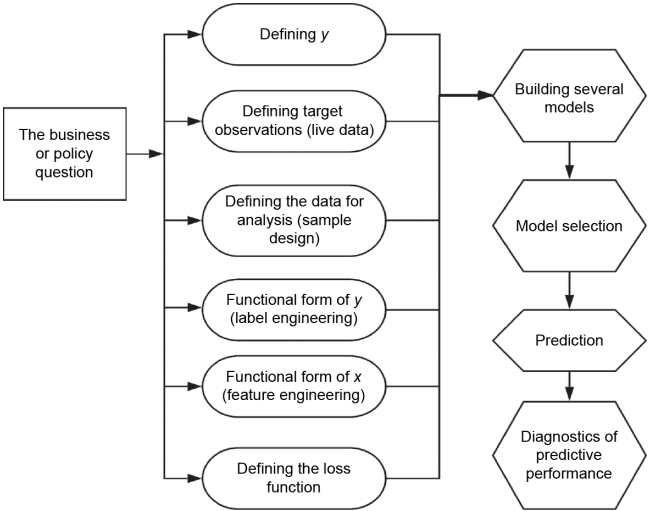


- Cambridge University Press, 2021
- **gabors-data-analysis.com**
  - Download all data and code: gabors-data-analysis.com/data-and-code/
- This slideshow is for **Chapter 14**

1. Business question and defining y

- The first task is defining the business or policy question we seek to answer.
  - What kind of car or hotel prices are we interested in?
  - How is our decision related to monetary or other rewards?
- Answers will guide any further action.
- Design process of the analysis

Steps of Prediction



Sample design

- In a prediction exercise, we are interested in predicting target for a set of units we care about and are less involved in generalization.
- The fact that we are less interested in inference and more in prediction makes working on our sample a bit more important.

Sample design: filtering

- Before settling on a model, we need to design the sample.
- Filtering our data to match the business/ policy question.
- It may involve dropping observations based on key predictor values,

Spotting errors

- For prediction exercise, we should spend more time on finding and deleting errors.
- We have no chance predicting extreme values, and certainly not errors.

Case study of used cars: Sample design

- dropping hybrid cars, manual gear, truck
- drop cars without a clean title (i.e., cars that had to be removed from registration due to a major accident)
- drop when suspect cars with clearly erroneous data on miles run,
- drop cars in a fair (=bad) condition, cars that are new
- Data cleaning resulted in 281 observations

Label engineering - defining target

- We need to define what will be our target variable.
- In some cases, this requires no action,
- Often it requires thinking and decision-making about definition.

- Binary vs continuous.
- Log vs level

Label engineering - log vs level

- When price is the target variable, its relation to predictor variables is often closer to linear when expressed in log price.
- Both substantive and technical reasons.
  - Log differences approximate relative, or percentage, differences, and relative price differences are often more stable.
- Keeping our target in level or transforming into a log value is an important modelling choice.
  - Not straightforward.

Label engineering - log vs level

- Importantly, when the target variable is expressed in log terms, we want to predict the value of the target variable ($\hat{y}$) not its log ($\widehat{\ln y}$).
- One may simply raise e to the power of the predicted log target variable : $\hat{y} = e^{\widehat{\ln y}}$.
- Not enough! One has to adjust this power by a function of the standard deviation of the regression residual $\hat{\sigma}$

$$\hat{y}_j = e^{\widehat{\ln y}_j} e^{\hat{\sigma}^2/2} \tag{1}$$

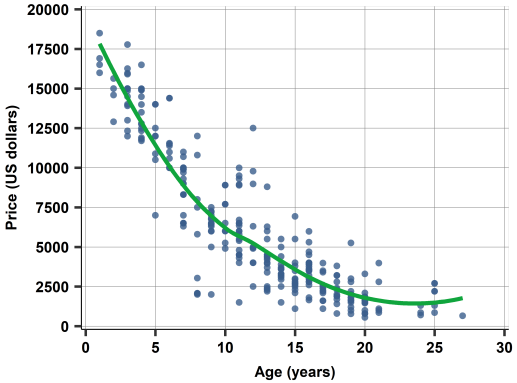
- Why correction term?
  - Regression predicts average (expected) ln *price*
  - We need average (expected) *price*
  - But average *exp*(*lnprice*) is not the same as average *price*
  - The ln function is concave

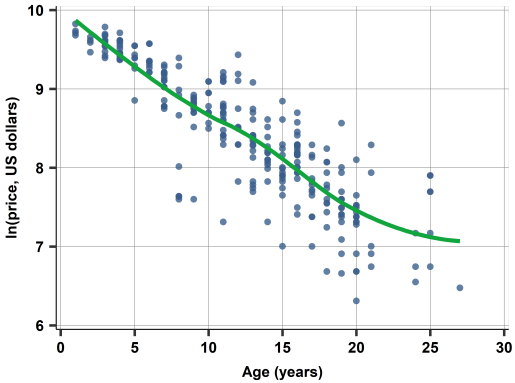
Label engineering - level or log?

- Business case is about price itself, continuous
- But model can have level or log price as target
- Log vs level model - some coefficients easier interpreted
- In this case, level is okay.

Label engineering - level or log?

Level with quadratic or linear in logs - both options of modelling seems okay.





Feature engineering: what x variables to have and in what functional form

- Feature engineering typically includes
  - deciding on what predictor variables to include
  - exploring and addressing potential missing values and extreme values
  - deciding about function forms of the predictor variables
  - designing possible interactions of the predictor variables
- Importantly, we use both domain knowledge - information about the actual market, product or the society - and statistics to make decisions.

What to do with different type of variables

- Type of variable
- If binary (e.g, yes/no; male/female; 1/2) – create a 0/1 binary variable
- If string / factor – check values, and create a set of binaries.
  - Often: key task will be to merge outcomes for the purpose of parsimony. Science + art
- Continuous – nothing to do. Make sure it is stored as number
- Text – Natural Language Processing. Mining the text to get useful info.
  - May simply go and find some words, and create binaries –> seminar
  - Everything else is complicated

Predicting Airbnb Apartment Prices - Intro

- Predicting Airbnb Apartment Prices: Selecting a Regression Model

- The goal is to predict the price that may be appropriate for an apartment with certain features.
- Business case: have apartments, need to price them for rent

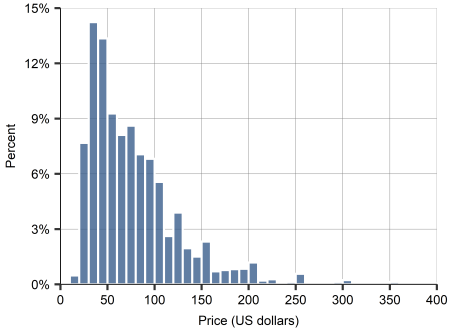
- London,UK
- http://insideairbnb.com
- 50K observations
- 94 variables, including many binaries for location and amenities
- Key variables: size, type, location, amenities
- Quantitative target: - price (in USD)

Airbnb prices

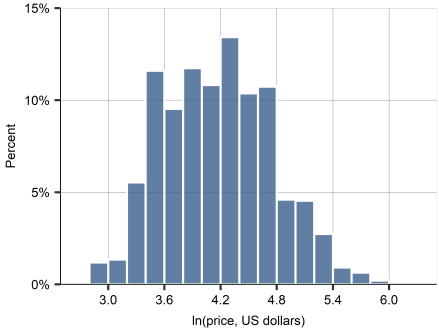
- Today: focus on a borough of Hackney
- N= 5K observations
- Price distribution (<400 USD)
  - Today: do it in level, could do logs
- Dropped very large apartments. Our final original data has 4393 observations.

Airbnb apartment price and ln price distributions

(a) Price



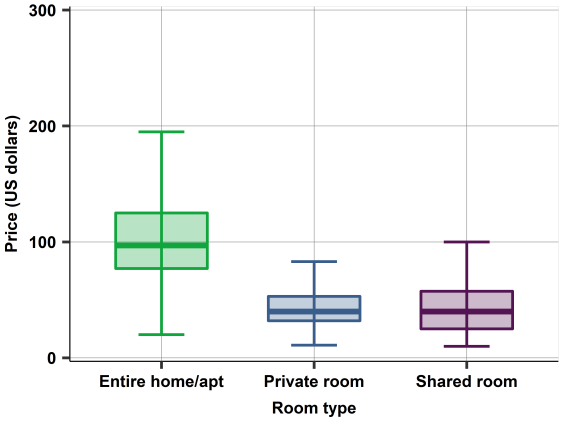
(b) Price in log



Note: Price for one night in US dollars. Histograms without extreme values (price $<$ 400 US dollars).

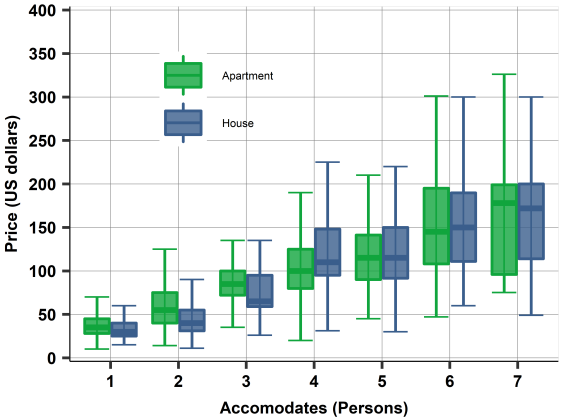
Airbnb apartment price distribution by important features

Price by room type



Airbnb apartment price distribution by important features

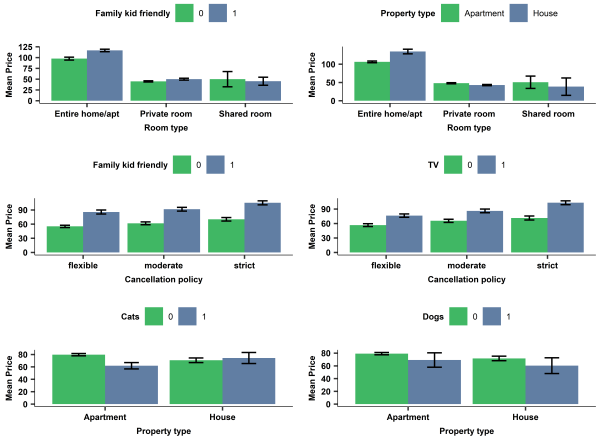
Price by number of people accommodated and apartment versus house



Feature engineering

- Key issue is to look at variables and think functional form
- Guests to accommodate goes up to 16, but most apartments accommodate 1 through 7. Keep as is. Add variables for type. No need for complicated models
- Regarding other predictors, we have several binary variables, which we kept as they were: type of bed, type of property (apartment, house, room), cancellation policy.
- Look at possible need for interactions by domain knowledge / visualization

Graphical way of finding interactions



Model building

- Model building is essentially deciding about the predictors to include in the model and their functional form.
- We have strong computers, cloud, etc - why could not we try out all possible models and pick the best one?

We Can't Try Out All Possible Models

- We have N observations and p predictors
- Main reason for model selection problem is that we can not try out every potential combination of models.
- As p increase, trying out all options becomes prohibitively complicated and computationally intractable.
- This types of problems in computer science are called **NP-hard**
- NP stands for "non-deterministic polynomial acceptable" problems.
- The consequence of this is rather important: There is no silver bullet in feature selection.

Model building

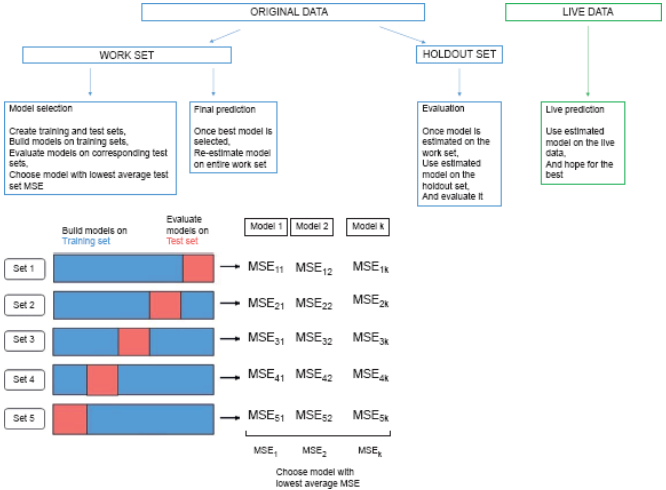
Two methods to build models:

- by hand - mix domain knowledge and statistics
  - Use domain knowledge drives picking key variables
  - Prefer variables that are easier to update - cheaper operation of a prediction model used in production
- by smart algorithms = machine learning

Evaluating the prediction using a holdout set

- Use various subsamples of the original data for various steps of predictive analysis
  1. Starting with the original data, split it into a larger work set and a smaller holdout set
  2. Further split the work set into training sets and test sets for k-fold cross-validation
  3. Build models and select the best model using k-fold cross-validation
  4. Re-estimate the best model using all observations in the work set
  5. Take the estimated best model and apply it to the holdout set
  6. Evaluate the prediction using the holdout set

Illustration of the uses of the original data and the live data



Selecting Variables in Regressions by LASSO

- Key question: which features to enter into model, how to select?
- There is room for an automatic selection process.
- Some are computationally very intensive (compare every option?)
- Advantage: no need to use outside info
- Disadvantage: may be sensitive to overfitting, hard to interpret

LASSO

- *LASSO* (the acronym of Least Absolute Shrinkage and Selection Operator) is a method to *select variables to include in a linear regression* to produce good predictions and avoid overfitting.
- It starts with a large set of potential predictor variables
- LASSO modifies the way regression coefficients are estimated by adding a penalty term for too many coefficients.

LASSO

Consider the linear regression with i=1...n observations and k variables, denoted 1...k:

$$y^E = \beta_0 + \beta_1 x_1 + ... + \beta_k x_k = \beta_0 + \sum_{j=1}^{k} \beta_j x_j \tag{2}$$

Coefficients are estimated by OLS: which minimizes the sum of squared residuals:

$$\min_{\beta} \left\{ \sum_{i=1}^{N} (y_i - (\beta_0 + \sum_{j=1}^{k} \beta_j x_{ij}))^2 \right\} \tag{3}$$

LASSO modifies this minimization by a penalty term:

$$\min_{\beta} \left\{ \sum_{i=1}^{N} (y_i - (\beta_0 + \sum_{j=1}^{k} \beta_j x_{ij}))^2 + \lambda \sum_{j=1}^{k} |\beta_j| \right\} \tag{4}$$

LASSO: how it works

- λ in the formula above is called the tuning parameter.
- It serves as a weight for the penalty term versus the OLS fit. Thus, it drives the strength of the variable selection
- Perhaps surprisingly, the main effect of this constraint is to force many coefficients to zero.

Airbnb Pricing Model building

- Process: build many models that differ in terms of features:
  - Which predictors are included
  - Functional form of predictors
- Here: specified eight linear regression models for predicting price.
- Data has 4393 observations. This is our original data.
  - 80% is our work set (3515 observations), the rest we will use for diagnostics.
- *Minor revision re textbook, small differences to numbers*

Versions of the Airbnb apartment price prediction models

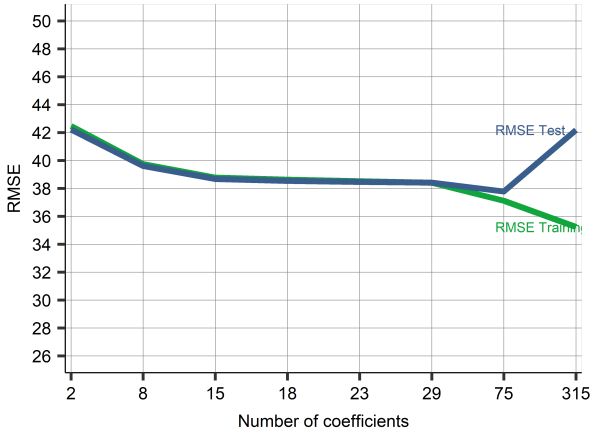
Model	Predictor variables	N var	N coeff
M1	guests accommodated as a quantitative variable, entered linearly	1	2
M2	= M1 + number of beds (linear), number of days since first review (linear), property type, room type, bed type	6	8
M3	= M2 + bathroom, cancellation policy, average review score (linear), number reviews (three categories), missing score flag	11	16
M4	= M3 + square term for guests, square and cubic terms for days since first review	11	17
M5	= M4 + room type and number of reviews interacted with property type	11	25
M6	= M5 + air conditioning and whether pets are allowed interacted with property type	13	30
M7	= M6 + all other amenities	70	87
M8	= M7 + all other amenities interacted with property type as well as bed type	70	315

Comparing model fit measures

Model	Coefficients	R-squared	BIC	Training RMSE	Test RMSE
M1	2	0.38	36 356	42.46	42.23
M2	8	0.46	35 941	39.74	39.61
M3	16	0.48	35 827	38.78	38.68
M4	17	0.49	35 824	38.63	38.53
M5	25	0.49	35 846	38.52	38.47
M6	30	0.49	35 877	38.41	38.42
M7	87	0.53	36 025	37.12	37.78
M8	315	0.56	37 679	35.24	42.19

Note: Average training set RMSE and average test set RMSE from 5-fold cross-validation for the eight regression models.

Training and test set RMSE for eight models



Training and test set RMSE for eight models

- Training RMSE falls with complexity
- Test RMSE falls then rises

- We pick Model M7 based on lowest CV RMSE..

The LASSO model

- Start with M8 and appr 300 candidate variables in the regression.
- We ran the LASSO algorithm with 5-fold cross-validation for selecting the optimal value for λ.
- LASSO regression just marginally better but: LASSO is automatic, a great advantage.
- Here: domain knowledge helped create M7. In other cases, LASSO could be great.

Post-prediction diagnostics

- Post-prediction diagnostics - understand better how our model works
- We look at prediction interval to learn about what precision we may expect to see of the estimates.
- We look at how the model work for different classes of observations
  - such as young and old cars.

Prediction with Big Data

- The principles of prediction are the same with Big Data as with moderate-sized data
- Big Data leads to smaller estimation error. This reduction makes the total prediction error smaller
- The magnitude of irreducible error, and problems with external validity, remain the same with Big Data
- When N is too large, we can take a random sample and select the best model with the help of usual cross-validation using that random sample

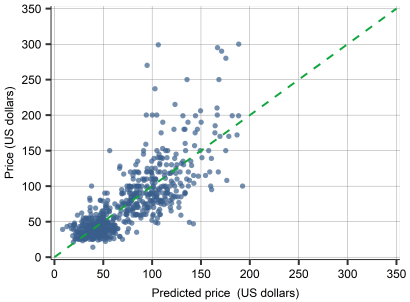
Data work and holdout

- Data has 4393 observations. This is our original data.
  - random 20% holdout set with 878 observations.
  - The remaining 80% is our work set (3515 observations).
  - Work set will be used for cross-validation with several folds of training and test sets.

Diagnostics

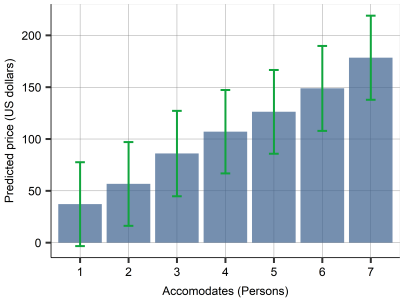
- Chose the OLS estimated M7.
- What can we say about model performance?
- After estimating the model on all observations in the work sample, we calculated its RMSE in the holdout sample. The RMSE for M7 is 41
- Higher than CV RMSE, could be other way around.
- Look at diagnostics on the holdout set.

Diagnostics: prices



- y-y-hat plot
- higher values not really caught.

Diagnostics: variation by size



- The model generates a very wide 80% PI for average apartment
- bar plot with PI bands
- wide intervals
- linear and thus, hurts small numbers more

Summary

- Our aim was to build a prediction model for pricing apartments
- We built a model, M7, with domain knowledge, and a horse race between models of various complexity
  - Picked the winner by cross-validated RMSE
- The model is useful for predication, but there is a great deal of uncertainty as suggested by diagnostics (on the holdout set)

Think external validity

- Future dataset will look different
- Think about how much
- Really matters in prediction
- If uncertain, pick simpler model

Main takeaways

- We can never evaluate all possible models to find the best one
  - Model building is important to specify models that are likely among the best
  - LASSO is an algorithm that can help in model building, by selecting the x variables and their functional forms
  - Exploratory data analysis and domain knowledge remain important alongside powerful algorithms, for assessing and improving the external validity of predictions