Regression & Prediction

A Journey Through House Prices

Your Name

xAI

February 23, 2025

Outline: Our Housing Journey

- Starting the Journey: Simple lines and fits
- Broadening the Picture: More clues and curves
- Refining Our Craft: Perfecting the story
- Navigating Pitfalls: Facing limits and flaws
- Bringing It Home: Examples and lessons

Starting the Journey

The Quest to Understand Relationships (Page 3)

- Imagine you're a detective, uncovering how house prices (Y) relate to size or location (X)
- Regression asks: "How does Y change with X, and can we predict it?"
- Bridges stats (past) and data science (future)

A Simple Start: Linear Regression (Page 4)

- Picture a line: $Y = b_0 + b_1 X$
- b_0: Start, b_1: Shift per X

```
# R (p. 152 adapted)
simple_lm <- lm(AdjSalePrice ~
    SqFtTotLiving, data = house)
```

figure4-2-placeholder.pd

Figure 1: Price vs. Size

Simple Start: Python

- Predicts price, errors as $Y - \hat{Y}$

```
# Python (p. 152 adapted)
from sklearn.linear_model import
    LinearRegression
predictors = ['SqFtTotLiving']
outcome = 'AdjSalePrice'
simple_lm = LinearRegression()
simple_lm.fit(house[predictors],
    house[outcome])
```

- First clue: Size matters

Finding the Best Fit: Least Squares (Page 5)

- Minimize the mess: $\sum(Y - \hat{Y})^2$
- Like a key in a lock—Legendre and Gauss's trick
- Fast, but outliers can throw us off

Broadening the Picture

More Clues: Multiple Linear Regression (Page 6)

- $Y = b_0 + b_1 X_1 + b_2 X_2 + \ldots$
- Size, lot, bedrooms twist the tale

```
# R (p. 152)
house_lm <- lm(AdjSalePrice ~
    SqFtTotLiving + SqFtLot +
    Bathrooms +
                Bedrooms + BldgGrade,
                     data = house)
```

- RMSE, R^2 enrich our map

Multiple Linear: Python

```
# Python (p. 152)
predictors = ['SqFtTotLiving', '
    SqFtLot', 'Bathrooms', 'Bedrooms
    ', 'BldgGrade']
house_lm = LinearRegression()
house_lm.fit(house[predictors],
    house[outcome])
```

- Output: $229 per sq ft

Categories Join the Tale: Factor Variables (Page 7)

- Townhouse or single-family? Categories
- Switches (0/1), baseline comparison

```r
# R (p. 164)
prop_type_dummies <- model.matrix(~
    PropertyType - 1, data = house)
```

- Cast grows richer

Factor Variables: Python

```
# Python (p. 166 adapted)
import pandas as pd
X = pd.get_dummies(house['
    PropertyType'], drop_first=True)
```

- Type shapes price

Curves in the Plot: Nonlinear Regression (Page 8)

- Small homes jump, mansions less—bends
- Polynomials (X^2), splines, GAMs

```
# R (p. 190)
library(splines)
knots <- quantile(house_98105$
    SqFtTotLiving, p = c(.25, .5,
    .75))
lm_spline <- lm(AdjSalePrice ~ bs(
    SqFtTotLiving, knots = knots,
    degree = 3) +
                SqFtLot + Bathrooms
                    + Bedrooms +
                    BldgGrade, data
                    = house_98105)
```

figure4-12-placeholder.p

Figure 2: Spline Fit

Nonlinear: Python

- Follows data's winding path

```
# Python (p. 190)
import statsmodels.formula.api as
    smf
formula = 'AdjSalePrice ~ bs(
    SqFtTotLiving, df=6, degree=3) +
    SqFtLot + Bathrooms + Bedrooms
    + BldgGrade'
model_spline = smf.ols(formula=
    formula, data=house_98105).fit()
```

- Not so straight anymore

Refining Our Craft

Checking Our Work: Model Assessment (Page 9)

- RMSE: How far off? R^2: How much captured?
- R and Python clue us in—hit the target
- Proofreading: Does it hold up?

Testing the Future: Cross-Validation (Page 10)

- Splits data into k pieces for tomorrow
- Train most, test one, repeat—future-ready
- Keeps us honest, not just past-fit

Picking the Best Story: Model Selection (Page 11)

- Trim clutter, balance with AIC
- Edit with Occam's razor—lean tale

```
# R (p. 157)
library(MASS)
house_full <- lm(AdjSalePrice ~
    SqFtTotLiving + SqFtLot +
    Bathrooms +
                  Bedrooms +
                      BldgGrade +
                      PropertyType,
                      data = house)
step <- stepAIC(house_full,
    direction = "both")
```

- Clear, predictive story

Weighing the Evidence: Weighted Regression (Page 12)

- Recent sales weigh more—trust fresher data
- Tweaks the tale since 2005

```
# R (p. 159)
house$Weight = year(house$
    DocumentDate) - 2005
house_wt <- lm(AdjSalePrice ~
    SqFtTotLiving + SqFtLot +
    Bathrooms +
                Bedrooms + BldgGrade,
                     data = house,
                    weight = Weight)
```

- Refines our focus

Navigating Pitfalls

Limits of Prediction: Extrapolation and Uncertainty (Page 13)

- Too far (empty lots) leads astray—stay in bounds
- Uncertainty: Confidence vs. prediction intervals
- Bootstrapping measures fuzziness

Decoding the Clues: Interpreting Coefficients (Page 14)

- Tricks: Correlation (size vs. bedrooms), multicollinearity
- Interactions (size in zips) add depth
- Read between lines—avoid twists

Spotting Flaws: Regression Diagnostics (Page 15)

- Outliers ($119,748), uneven errors hint flaws
- Focus: What works over perfection

```
# R (p. 177)
house_98105 <- house[house$ZipCode
    == 98105, ]
lm_98105 <- lm(AdjSalePrice ~
    SqFtTotLiving + SqFtLot +
    Bathrooms +
                Bedrooms + BldgGrade,
                     data = house_
                    98105)
sresid <- rstandard(lm_98105)  #
    -4.326732
```

figure4-6-placeholder.pd

Figure 3: Influence Plot

Bringing It Home

The Story in Action: Housing Examples (Page 16)

- Linear ties price to size, splines curve reality
- Diagnostics: \$119,748 partial sale sways line
- Real predictions, data's quirks

Lessons from the Journey (Page 17)

- Bends from lines to curves—adaptable
- RMSE, cross-validation chase prediction
- R & Python fuel data stories

The Road Ahead (Page 18)

- Deepen: *Statistical Learning*, *Time Series with R*
- Next: Splines, time series
- Blend art and science—keep refining