Stat 106

Cullen MacNeil

General Analytic & Sports Metrics

Expected Points (EP)

- Average points scored given a specific situation or state in a game.
- Calculated based on historical play-by-play data.
- Example: Football EP at given yard line and down.

Expected Points Added (EPA)

- Measure of value added by a play relative to expectation.
- Formula:

$$EPA = Points - EP$$

- Interpretation: Positive EPA indicates better-than-expected outcomes.

Win Probability (WP) & Win Probability Added (WPA)

- **Win Probability (WP)**: Likelihood of winning given the current game state.
- **Win Probability Added (WPA)**: Change in WP from before to after a specific play.
- Formula:

$$WPA = WP_{\text{after}} - WP_{\text{before}}$$

- High leverage situations (e.g., late-game) significantly affect WPA.

Core Analytical Ideas

Stickiness, Leverage, Clutch-ness

- **Stickiness**: Stability of performance metrics over time.
- **Leverage**: Situational importance; high-leverage moments significantly influence outcomes.
- **Clutch-ness**: Ability to perform well in high-leverage situations.

Luck & Mean Reversion

- **Luck**: Random deviations from expected performance metrics.
- **Mean Reversion**: Tendency for extreme performance to return toward average levels over time.

Shrinkage Estimates

- Estimates adjusted towards a prior or mean to reduce variance.
- Useful for stabilizing performance estimates, particularly with limited data.
- Prevents overfitting and extreme predictions.

Regression Models

Linear Regression

- Model form: $Y = \beta_0 + \beta_1 X + \epsilon$.
- Interpretation: Coefficient β_1 is the average change in Y per unit change in X.
- Assumptions: linearity, independence, homoscedasticity, normality.

Logistic Regression

- Used for binary outcomes; models log-odds of an event.
- Model form: $\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 X$.
- Coefficients represent log-odds; e^{β} gives odds ratios.

Linear vs Logistic

- **Linear**: Continuous response variable.
- **Logistic**: Binary response variable.
- Choose based on outcome type (continuous vs. categorical).

Transformations & Interactions

Log Transformations

- Useful to stabilize variance and normalize skewed data.
- Often used when relationships between variables are multiplicative.

Polynomial Transformations

- Capture non-linear relationships using powers of predictors (e.g., quadratic, cubic).
- Example: $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \epsilon$.

Interaction Effects

- Effect of one predictor on the response depends on another predictor.
- Modeled by multiplying two predictors: $Y = \beta_0 + \beta_1 X + \beta_2 Z + \beta_3 (XZ) + \epsilon$.

Simulation & Resampling

Bootstrapping

- Repeated resampling from observed data to estimate sampling distribution.
- Useful for confidence intervals, bias estimation, and variance reduction.

Permutation Testing

- Hypothesis testing by reshuffling labels to determine significance.
- No assumptions about underlying distribution required.

Rating & Ranking Systems

Bradley-Terry Models

- Models probability of winning pairwise comparisons.
- Probability team A beats team B:

$$P(A \text{ beats } B) = \frac{e^{\beta_A}}{e^{\beta_A} + e^{\beta_B}}$$

- Estimates team strength parameters β via logistic regression framework.

Bradley-Terry (Quantitative)

- Extends Bradley-Terry model to predict numeric outcomes.
- Suitable for scores or other continuous metrics instead of binary win-loss outcomes.

Elo Models

- Dynamic rating system updating ratings based on outcomes.
- Win Probability formula:

$$P = \frac{1}{1 + 10^{(R_B - R_A)/400}}$$

- Update formula:

$$R_{\text{new}} = R_{\text{old}} + k(\text{Outcome} - \text{Expected})$$

- Parameter k adjusts sensitivity to new results.

KenPom Efficiency

- Basketball-specific metrics evaluating team efficiency.
- Offensive and defensive ratings adjusted for opponent strength and pace.
- Higher efficiency indicates stronger overall performance.

Predictive Modeling & Validation

Train-Test-Validation Split

- Data partitioned into subsets:
  - **Training**: Model fitting.
  - **Validation**: Model selection and tuning.
  - **Test**: Evaluate out-of-sample predictive performance.

Cross-Validation

- Technique to assess model predictive performance.
- Data repeatedly split into training and validation subsets.
- Commonly used form: k-fold CV, data split into k subsets.

Prediction, Overfitting, Complexity

- **Overfitting**: Model captures noise, poor generalization.
- Balance complexity (number of parameters) vs. prediction accuracy.
- Cross-validation helps identify appropriate model complexity.

Model Selection & Regularization

Sequential Variable Selection (AIC)

- Iteratively adds/removes variables based on Akaike Information Criterion (AIC).
- AIC formula: $AIC = 2p - 2\ln(L)$, penalizes model complexity.

Penalized Regression

- Adds penalty term to regression to prevent overfitting.
- **Ridge Regression**: Penalizes squared coefficients $\sum \beta_j^2$.
- **LASSO Regression**: Penalizes absolute value of coefficients $\sum |\beta_j|$, shrinks some coefficients to exactly zero.

Advanced Predictive Techniques

Random Forests

- Ensemble of decision trees built on bootstrapped samples.
- Reduces variance and improves prediction by averaging outcomes.
- Each split considers random subset of predictors.

Hyperparameter Tuning

- Process of optimizing model parameters not learned from data.
- Common methods: Grid Search, Random Search, Cross-Validation.
- Helps prevent overfitting and improves predictive performance.

Additional Quick Reference

Regression Assumptions

OLS Linear Regression:

- Linearity, independence, homoscedasticity, normality of residuals.

Logistic Regression:

- Independence, linearity in logit scale.

Important Metrics & Formulas

Pythagorean Wins:

$$E(\text{Wins}) = G \times \frac{RS^2}{RS^2 + RA^2}$$

Confidence Interval (mean):

$$\bar{x} \pm t^* \frac{s}{\sqrt{n}}$$

Z-Test Statistic:

$$Z = \frac{\bar{x} - \mu_0}{\sigma/\sqrt{n}}$$

Essential R Functions

dplyr (Tidyverse):

- `filter(data, condition)`
- `select(data, columns)`
- `mutate(data, new_var = expression)`
- `summarise(data, new_var = summary_function)`

Regression and Trees:

- Linear Regression: `lm(y ~ x, data)`
- Logistic Regression: `glm(y ~ x, data, family = "binomial")`
- Regression Trees: `rpart(y ~ x, data)`
- Random Forests: `randomForest(y ~ x, data)`

Model Interpretation

- Linear regression coefficient: change in Y per unit change in X.
- Logistic regression coefficient: log-odds, e^β for odds ratio.
- Adjusted R^2: accounts for number of predictors.

Quick Visualization (ggplot2)

- Histogram: `geom_histogram()`
- QQ plot: `stat_qq(), stat_qq_line()`
- Scatterplot: `geom_point(), geom_smooth(method = "lm")`