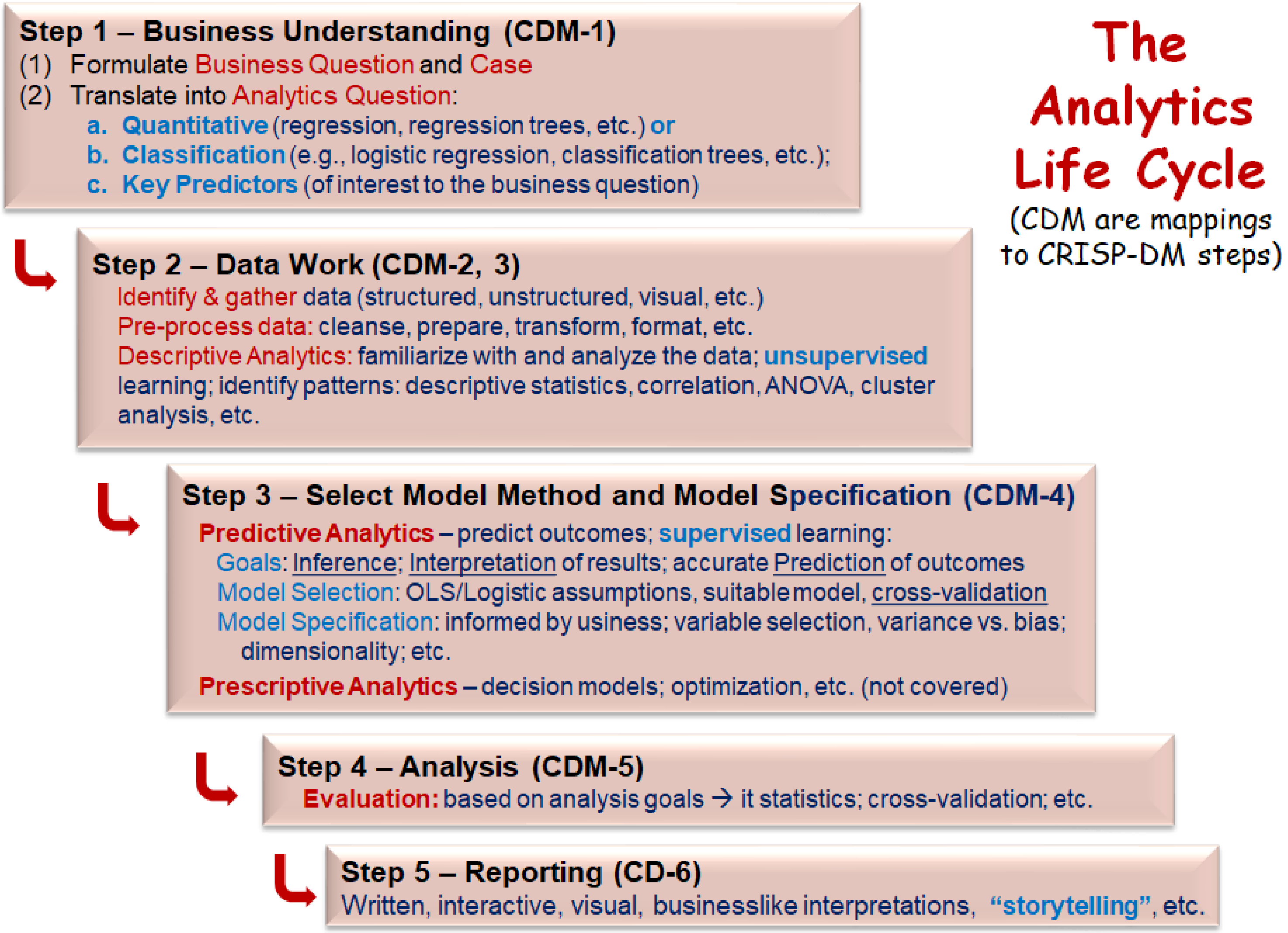
ITEC 621 Predictive Analytics

Predictive Model and Method Summary

Prof. Alberto Espinosa *(last updated 4/2/2022)*



The Analytics Life Cycle

(CDM are mappings to CRISP-DM steps)

Step 1 – Business Understanding (CDM-1)

(1) Formulate Business Question and Case

(2) Translate into Analytics Question:

a. **Quantitative** (regression, regression trees, etc.) or

b. **Classification** (e.g., logistic regression, classification trees, etc.);

c. **Key Predictors** (of interest to the business question)

Step 2 – Data Work (CDM-2, 3)

Identify & gather data (structured, unstructured, visual, etc.)

Pre-process data: cleanse, prepare, transform, format, etc.

Descriptive Analytics: familiarize with and analyze the data; **unsupervised** learning; identify patterns: descriptive statistics, correlation, ANOVA, cluster analysis, etc.

Step 3 – Select Model Method and Model Specification (CDM-4)

Predictive Analytics – predict outcomes; **supervised** learning:

Goals: Inference; Interpretation of results; accurate Prediction of outcomes

Model Selection: OLS/Logistic assumptions, suitable model, cross-validation

Model Specification: informed by usiness; variable selection, variance vs. bias; dimensionality; etc.

Prescriptive Analytics – decision models; optimization, etc. (not covered)

Step 4 – Analysis (CDM-5)

Evaluation: based on analysis goals → it statistics; cross-validation; etc.

Step 5 – Reporting (CD-6)

Written, interactive, visual, businesslike interpretations, **"storytelling"**, etc.

Analytics Model and Specification Selection Process

1. Review and carefully understand your **analytics question(s)**. This involves understanding:

   a. Your response variable(s) – what are you trying to predict?

   b. Whether you have a quantitative or classification problem – note: this may change as you carry out your analysis, as we sometimes transform quantitative variables into categorical and vice versa.

   c. What are some of the likely predictors you may use – naturally, you are only speculating at the beginning, but you should have some idea from a business perspective about what are the possible predictors based on business knowledge or intuition.

2. Identify, review and **describe your data**:

   a. Inspect your data sources for omitted data and inconsistencies

   b. Do the necessary descriptive analytics review, including but not limited to: descriptive statistics, correlation matrix, correlation plots, other plots to evaluate things like normality, linearity, etc. No need to do everything, but only what is needed to do your predictive modeling

   c. Do all the necessary data pre-processing and transformations – some needed transformations may not be obvious until later

3. Try first, either:

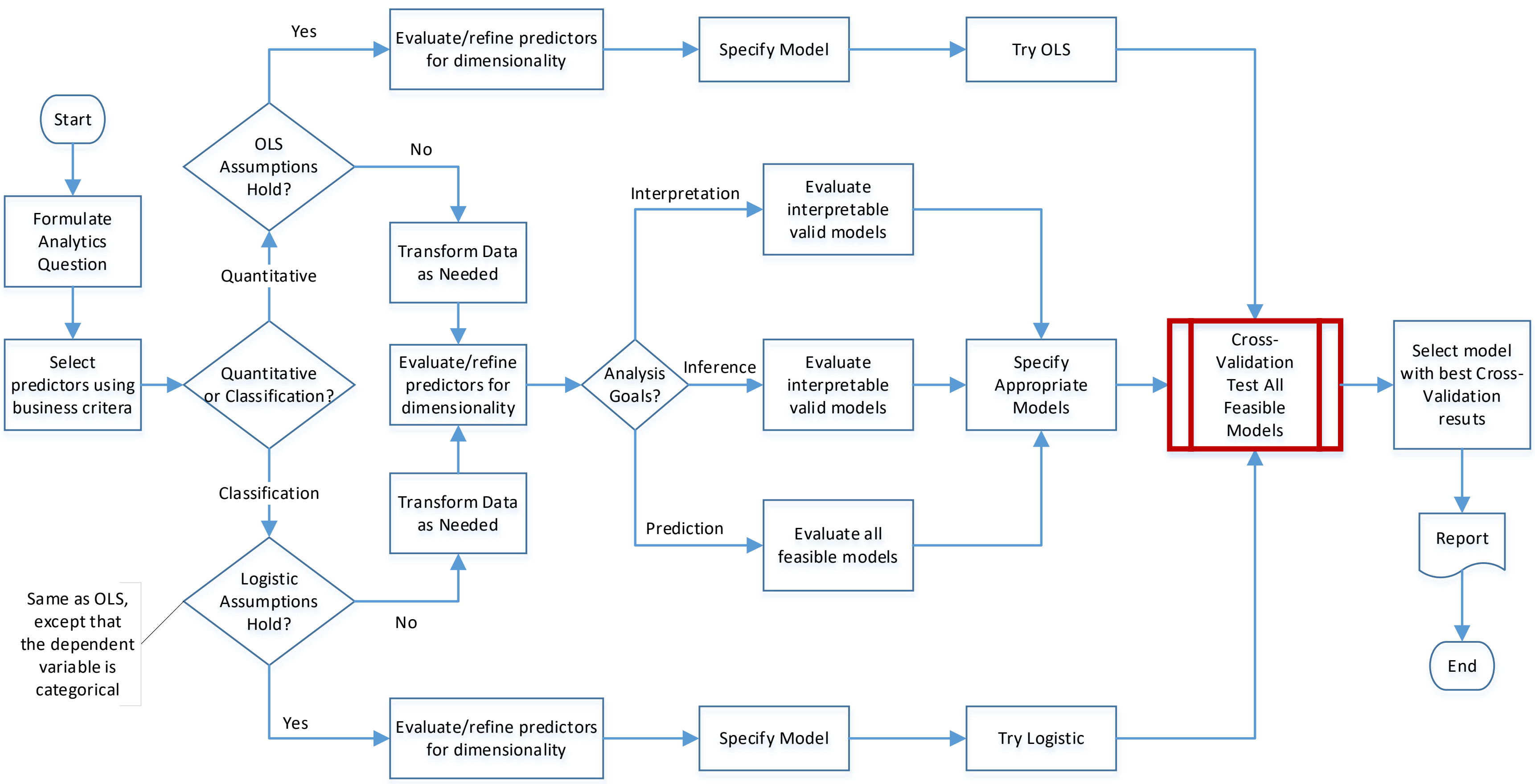
   a. An **OLS** model, if you are addressing a quantitative problem.

   b. A **Logistic** model, if you are addressing a classification problem.

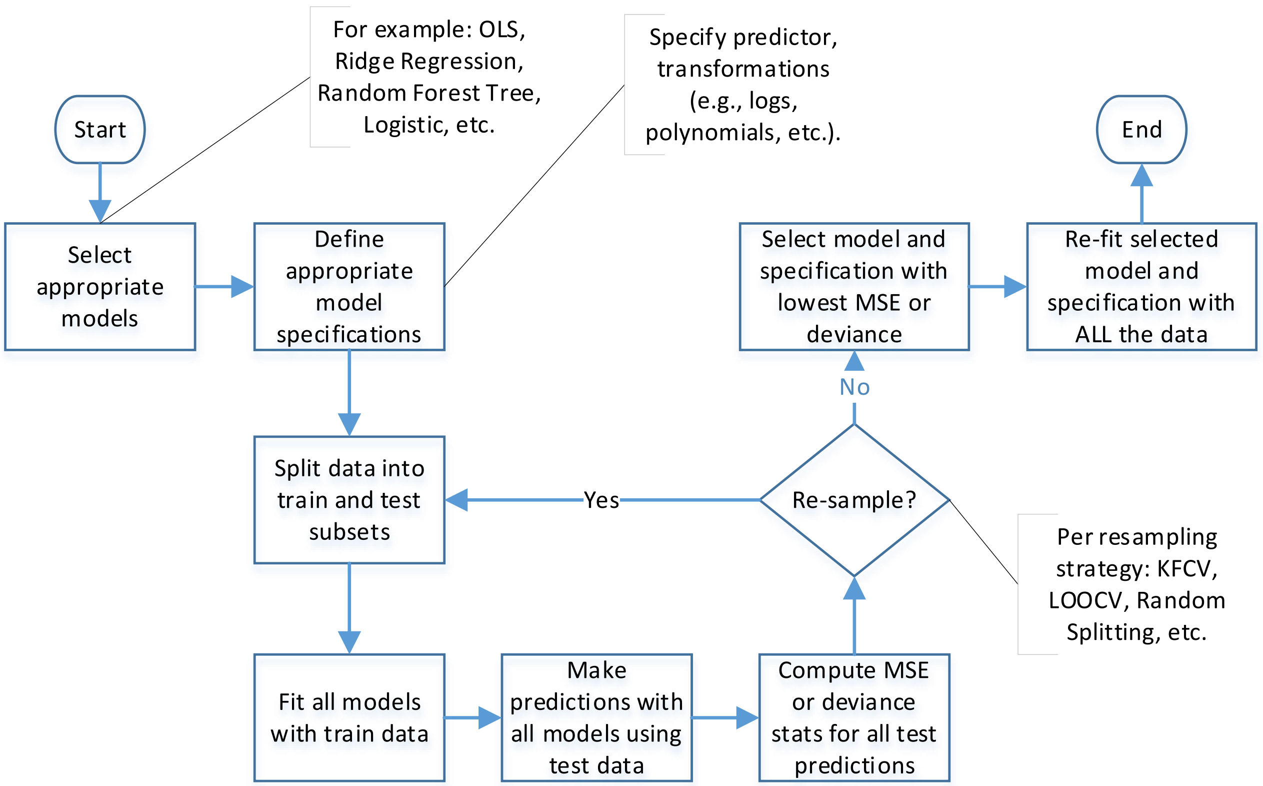
   c. Use the full data set on your first exploratory OLS or Logistic models

4. If you have many predictors, you may want to rationalize the **best subset** of predictors from a business perspective. You should then determine the appropriate number of predictors using a variable selection method.
5. **Test** key **OLS assumptions** as needed – e.g., normality of the response variable, multi-collinearity, heteroscedasticity, serial correlation (only for time related data), linear relationship between predictors and the response variable, etc. – OLS is BLUE if all assumptions hold (Logistic is BLUE for classification problems).
6. **Correct** for OLS assumption violations, as needed, when possible (e.g., transformations, variable selection, weighting, etc.)
7. State your **analysis goals** – **interpretation** (explanation of effects), **inference** (hypotheses testing), and/or **prediction**; and narrow down your choice of models accordingly.
8. **Fit** the selected models using the full data set and **evaluate** these viable models using: (a) **fit statistics** (e.g., adjusted R^2, MSE, AIC, error rates, etc.) and then **cross-validation** of all the viable models by sampling/re-sampling training data to fit the model and computing fit statistics (i.e., MSE, classification error, deviance, sensitivity, specificity, ROC curves, etc., as appropriate) on the test data. Please rationalize and justify your cross-validation and re-sampling method (i.e., subset partition, K-Fold validation or Leave One Out validation).
9. **Select** the model with **best** cross-validation **test results**. For similar cross-validation results, select the most **interpretable** model
10. **Re-Fit** the **selected model**, but this time using the **full dataset**

Analytics Model and Specification Selection Process


Start
Formulate Analytics Question
Select predictors using business critera
Quantitative or Classification?
Quantitative
OLS Assumptions Hold?
Yes
Evaluate/refine predictors for dimensionality
Specify Model
Try OLS
No
Transform Data as Needed
Evaluate/refine predictors for dimensionality
Classification
Logistic Assumptions Hold?
Same as OLS, except that the dependent variable is categorical
No
Transform Data as Needed
Yes
Evaluate/refine predictors for dimensionality
Specify Model
Try Logistic
Analysis Goals?
Interpretation
Evaluate interpretable valid models
Inference
Evaluate interpretable valid models
Prediction
Evaluate all feasible models
Specify Appropriate Models
Cross-Validation Test All Feasible Models
Select model with best Cross-Validation resuts
Report
End

Machine Learning → Cross Validation Process


Start
Select appropriate models
For example: OLS, Ridge Regression, Random Forest Tree, Logistic, etc.
Define appropriate model specifications
Specify predictor, transformations (e.g., logs, polynomials, etc.).
Split data into train and test subsets
Fit all models with train data
Make predictions with all models using test data
Compute MSE or deviance stats for all test predictions
Re-sample?
Per resampling strategy: KFCV, LOOCV, Random Splitting, etc.
Yes
No
Select model and specification with lowest MSE or deviance
Re-fit selected model and specification with ALL the data
End

Modeling Method, Model Specification and Procedures

Analytics Question	Model Types	Modeling Methods (Evaluate 3)	Model Specification	Testing, Procedures and Comments
Quantitative	Parametric	• Regression: OLS, Ridge, LASSO, PCR, PLS	• Predictors (feature) selected based on business knowledge • Predictors (features) selected by statistical analysis (e.g., variable selection, ANOVA, stepwise, etc.) • Variable transformations (e.g., polynomials, log models, lagged or time-series models, piecewise models, splines, ranks, Box-Cox transformations, etc.) • Categorical to binary variable transformations are not different specifications but required transformations. • Centering and standardizing are not model specifications, but different interpretations of results.	• WLS is not really a different method, but a correction for OLS' heteroskedasticity. • GLM is not really a different method, but an estimation method applicable to several methods. • Step variable selection (Forward, Backward and Stepwise) are not model methods, but evaluation procedures for variable selection. • Bootstrap, RSCV, LOOCV and KFCV are not model methods, but cross-validation tests applicable to any method. • You can use R^2, Adjusted R^2, model p-value, coefficient p-values, Cp, 2LL, AIC, BIC, etc. to analyze the statistical fit. But all candidate models and specifications must be compared with a cross-validation testing method. • Changing tuning parameters don't yield different models or specifications but help select the optimal tuning of models (e.g., tree size, Ridge shrinkage, PCR components).
	Non-Parametric	• Regression trees, bootstrap aggregation trees, random forest trees, boosted trees, neural networks, k-nearest neighbors		
Classification	Parametric	• Binomial and multinomial Logistic regression • LDA and QDA		
	Non-Parametric	• Classification trees, bootstrap aggregation trees, random forest trees, boosted trees, support vector machines, neural networks		

Note: only models and specifications covered in this course are allowed for the course project. Any team contemplating other methods not covered in the course must obtain permission from the instructor.

Model and Specification Selection Summary Table

Notation: ***Type:** R or T (Regression or Tree)/Q or C (Quantitative Value or Classification Outcome)
**** OLS Assumptions:** (✓) Holds; (×) Does not Hold;
***** Notation:** Y = Outcome Variable; X's = Predictor Variables; N = Number of Observations; P = Number of Predictors

Method/Model	***Type**		**** OLS Assumptions – use when:**	**Test**	***** When to Use/Comments**
Models Covered in Class					
Ordinary Least Squares (OLS)	R	Q	1. YC (✓) Y is continuous 2. EN (✓) Errors are normally distributed 3. XI (✓) X's are independent (uncorrelated) 4. LI (✓) Y and X's have linear relationship 5. OI (✓) Observations are independent 6. EI (✓) Errors are independent 7. EA (✓) The error average is 0 8. EV (✓) The error variance is constant	• Test for compliance with OLS assumptions. • See tests below	• If OLS assumptions hold → **OLS is BLUE** (best linear unbiased estimator → Gauss-Markov Theorem) • It means it is the most efficient → Lowest variance • Need to test the assumptions • Excellent for **inference/interpretation** (same is true for most regression models) • Good for **prediction**, but other models with smaller variance may have better cross-validation results • Need large samples with 30+ degrees of freedom (i.e., N-P-1 > 30)
Transformations			May or may not comply with OLS assumptions.	• Inspect the data	• Some transformations are needed to address non-compliance with OLS assumptions (e.g., Log(Y), Rank(Y), Lag(Y or X's), etc. • Other transformations are for convenience, improve statistical fit and predictive accuracy, or to quantify categorical data (e.g., Log(X's), Rank(X's), categorical to binary data.
Weighted Least Squares (WLS)	R	Q	EV (×) The error variance is not constant	• Breusch-Pagan or White tests for heteroskedasticity	• OLS is unbiased, but the variance is wrong • WLS is more efficient • Use when heteroscedasticity is present (i.e., EV does not hold – i.e., error variance varies with Y • Good for **inference/interpretation** and **prediction**
Standardized OLS	R	Q	All OLS assumptions (✓)	• No test, if OLS is OK, Std. OLS is OK too	• Use when scale of X's vary and you want comparable predictors • Or when scale of X's not easily interpreted (e.g., survey ratings) • Good for **inference/interpretation** and **prediction**
Log Transformed OLS	R	Q	EN (×) Residuals are not normally distributed. If Y is not normally	• Test Y for normality with histogram, qq-	• If Y has a known distribution (e.g., binary) use GLM with the appropriate distribution family.

			distributed, the residuals are probably not normally distributed too. Logs often help correct for skewed distributions.	plot and Shapiro-Wilk test. • If Y is not normal, test the residuals for normality. • Test X's for normality if sample is small.	• If Log(Y) is normal, you can just log-transform Y. • You can log-transform some X's if you are interested in the effect of a % increase, rather than a unit increase • You must log-transform X's if not normally distributed and the sample is small (no need for large samples) • Can't log transform Y or any X's if they contain negatives or 0 (no logs) • Good for **inference/interpretation** and **prediction**
Log Transformed GLM	R	Q	EN (×) Residuals are not normally distributed. If Y is not normally distributed, the residuals are probably not normally distributed too. GLM can be use for a wide range of distributions of Y (e.g., logit, counts) YC (×) Y is not continuous	• Same as above	• Use when the response variable contains **count data** or is **binary** • Data is discrete (not continuous); truncated at 0; and with uneven errors (low near 0 and increasing as counts get larger • A popular model for count data is to use the Generalized Linear Method (**GLM**) with a **log-transformed Y** and a **Poisson** distribution.
Rank Transformed OLS	R	Q	EN (×) Errors are not normally distributed (e.g., distribution is skewed) Note: any rank-transformed variable becomes non-parametric and has a uniform distribution	• Same as above.	• Useful with small samples • Also, when data is not normally distributed and the distribution doesn't seem to have a pattern • And when there is very little variance in a variable. • OLS assumptions no longer apply to rank transformed variables • OK for **inference/interpretation**; good for **prediction**
Lagged Models	R	Q	EI (×) Errors are not independent → they vary systematically across time (e.g., serial correlation)	• Durbin-Watson or Breusch-Godfrey tests for serial correlation.	• Typically used when the X's include a time variable • And when the **Durbin-Watson** test finds auto correlated residuals (e.g., serial correlation) • Good for **inference/interpretation** and **prediction**
Ridge Regression	R	Q	XI (×) X's are not independent (correlated)	• Test for Multicollinearity • Condition Index CI>30 not good, but tolerable; CI>50 is severe. • Variance Inflation Factors VIF >5 not good but tolerable; VIF>10 is severe; high	• Even if multicollinearity is not present, Ridge is an attractive model when dimensionality is high (e.g., hundreds of variables). • Increases predictor **bias**, but reduces **variance** • But provide better predictive accuracy and cross validation when there are **too many predictors** (i.e., curse of dimensionality). • A desirable method when there are too many predictors in the model and the goal is **accurate predictions**

				VIF predictors are the main contributors to multicollinearity	• And when it is important to **keep all predictors** • Coefficients are shrunk, but they don't become 0 • How much shrinkage will depend on the selected tuning parameter λ (**=0 → OLS**; **=∞ → null model**, i.e., no coefficients, just intercept) • OK for **inference/interpretation**; good for **prediction**
LASSO Regression	R	Q	XI (×) X's are not independent (correlated)	• Same as above	• Same as above, but LASSO is better than Ridge if coefficients can shrink to 0; if not Ridge is better. • With LASSO, some coefficients do become 0 • OK for **inference/interpretation**; good for **prediction**
Principal Components Regression (PCR)	R	Q	XI (×) Some X's are not independent (correlated)	• Same as above	• Useful when there is a **large number** of **predictors P** • And there is **high multi-collinearity** and multiple variables that appear to measure similar things (e.g., size of car, weight of car) • Useful when we want to weight the effect of all predictors, but we are not so interested in the effect of specific predictors, just all of them • **Not** so **good** for **inference/interpretation**; good for **prediction**
Partial Least Squares (PLS)	R	Q	XI (×) Some X's are not independent (correlated)	• Same as above	• Same as above. • PLS is an alternative method to PCR. • The best thing is to use both and select the one with smallest test cross validation fit statistics.
Interaction Models	R	Q	LI (×) Y does not have a linear relationship with all X's	• There is no formal test to recommend interactions. • Modeling interaction terms is more of an exploratory approach. • Use business domain knowledge to assess if two predictors may interact	• Useful when you suspect that the value of one predictor influences the effect of another predictor (e.g., the effect of an antibiotic is diminished if you drink alcohol). • Very good for **inference/interpretation**; very good for **prediction**
Polynomial Models	R	Q	LI (×) Y does not have a linear relationship with all X's	• Use if scatter plots of Y with the X's and inspect if the	• Quadratic, Cubic and other polynomial models will fit the data better.

				relationship does not appear to be linear.	• The value for **inference/interpretation** diminishes as the power of the polynomial goes up; but the **prediction** value goes up (except at both tail ends). • High polynomials suffer from **over-identification** and **high variance**
Step Regression Models	R	Q	LI (×) Y does not have a linear relationship with all X's	• Same as bove	• Good when data shows **different patterns** in different **sections** • Preferred to polynomial models when polynomials don't fit the data well at both ends of the curve (i.e., waging the tail) • Regression lines are **horizontal** and they change the intercept in different sections of the data • Good for **inference/interpretation** but the value diminishes as the number of steps increases; but the **prediction** value goes up.
Piecewise Models	R	Q	LI (×) Y does not have a linear relationship with all X's	• Same as above	• Similar to Step models, but more effective when the **data** in each section appears to **slope linearly** upwards or downwards. • Good for **inference/interpretation** but the value diminishes as the number of piecewise sections increases; but the **prediction** value goes up. It generally performs better than polynomial regression at the tails.
Spline Regression (MARS)	R	Q	LI (×) Y does not have a linear relationship with all X's	• Same as above	• Similar to Piecewise models, but more effective when the data appears to have a curvilinear relationship with Y • Not good for **inference/interpretation** especially as the power of the spline goes up; very good for **prediction.** It generally performs better than polynomial regression at the tails.
Binomial Logistic Regression	R	C	YC (×) Y is not continuous EN (×) Errors are not normally distributed	• No test needed. It is always obvious if Y is a binary classification.	• Most popular regression model when Y is binary (e.g., yes/no, approve/decline, etc.) • Y can only have **two classes** (hence **binary**) • It tends to outperform LDA and QDA below when observations don't come from a normal distribution • The model predicts the probability of falling into one of the classes, not the actual value of Y

					• OK for **inference/interpretation** if the goal is to interpret likelihoods or probabilities, not values; good for **prediction**
Multinomial Logistic Regression	R	C	YC (×) Y is not continuous EN (×) Errors are not normally distributed	• Same as above	• Use instead of Binary Logistic Regression when Y has more than two classes (e.g., Rural, Suburban, Urban)
Linear/Quadratic Discriminant Analysis (LDA/QDA)	R	C	YC (×) Y is not continuous EN (×) Errors are not normally distributed	• Same as above • But also test the X's for normality	• If N is large, Logistic Regression and Discriminant Analysis produce similar results, but Logistic is preferred because the model is more interpretable. • If **N** is **small** and **X'**s are **normally distributed** Discriminant Analysis models are more stable and Discriminant Analysis is preferred if **prediction** is the goal and **inference/interpretation** is not.
Regression Trees	T	Q	Non-parametric. OLS assumptions and other parametric restrictions don't apply	• No tests needed.	• Select tree methods based on lowest test cross-validation MSE • Not good for **inference/interpretation**; may be good for **prediction** depending on test cross-validation statistics.
Classification Trees	T	C	Non-parametric. OLS assumptions and other parametric restrictions don't apply	• No tests needed	• Same as above but compare models based on lowest test cross-validation deviance. • Not good for **inference/interpretation**; may be good for **prediction** depending on cross-validation statistics.
Bootstrap Aggregation (Bagging)	T	C	Non-parametric. OLS assumptions and other parametric restrictions don't apply	• No tests needed	• Same as above • Bagging tends to outperform regular tree methods
Random Forests	T	C	Non-parametric. OLS assumptions and other parametric restrictions don't apply	• No tests needed	• Same as above • Bagging tends to outperform regular tree methods

Models Not Covered in Class, see Goodies					
Smoothing Splines	R	Q	Same as Polynomial Models	• Same as polynomials	• Similar to Spline models, but the transition of the curves from one section to another is "smoothed" • Not good for **inference/interpretation**; very good for **prediction** and it generally performs better than polynomial regression at the tails.
K Nearest Neighbors (KNN)	na	C	Non-parametric. OLS assumptions and other parametric restrictions don't apply	• Non-parametric method • No tests needed	• Generally outperforms logistic and LDA/QDA when the decision non-linear and has no clear pattern. • Not good for **inference/interpretation**; may be good for **prediction** depending on cross-validation statistics.
Boosted Trees	T	C	Non-parametric. OLS assumptions and other parametric restrictions don't apply	• Same as for other tree methods	• Not good for **inference/interpretation**; much better than classification or regression trees for **prediction** but cross-validation statistics need to be checked to select the model with highest predictive accuracy.
Support Vector Machines	T	C	Non-parametric. OLS assumptions and other parametric restrictions don't apply	• Non-parametric method • No tests needed.	• Not good for **inference/interpretation**; much better than classification or regression trees for **prediction** but cross-validation statistics need to be checked to select the model with highest predictive accuracy.
Neural Networks	R	Q	XI (×) Some X's are not independent (correlated)	• Same as PCR and PLS	• Useful when there is a very large number of predictors P • And there is high multi-collinearity and multiple variables that appear to measure similar things (e.g., size of car, weight of car) • Not good for **inference/interpretation**; good for **prediction**
Structural Equation Models	R	Q	XI (×) Some X's are not independent (correlated)	• Same as PCR and PLS	• Useful when multi-collinearity is very or extremely high, but it is important to retain all or most predictors in the model. • Or when structural relations are hypothesized (e.g., some variables predict some outcomes, and these outcomes predict other outcomes in turn, and so on. • Very good for **inference/interpretation**; good for **prediction**