

Business Data Analysis

MGM509

Lecture 3

Building Better Models: Regression,
SVMs, and Model Fitting

Prepared by
Eng. Ahmed Hussein, MBA, PfMP®,
PMP®, CMP APMG

Lecture
3

What we will cover

1. Data Preparation (Steps, sources, formats, tools, and storage of data)
2. What is Model Fitting?
3. Regression Types and How to choose the suitable Model

Lecture
03

Data Preparation



Lecture
3

Data Mining Process Overview (CRISP-DM)

- Cross Industry Standard Process for Data Mining
- Not strictly sequential; often involves **loops and iteration**

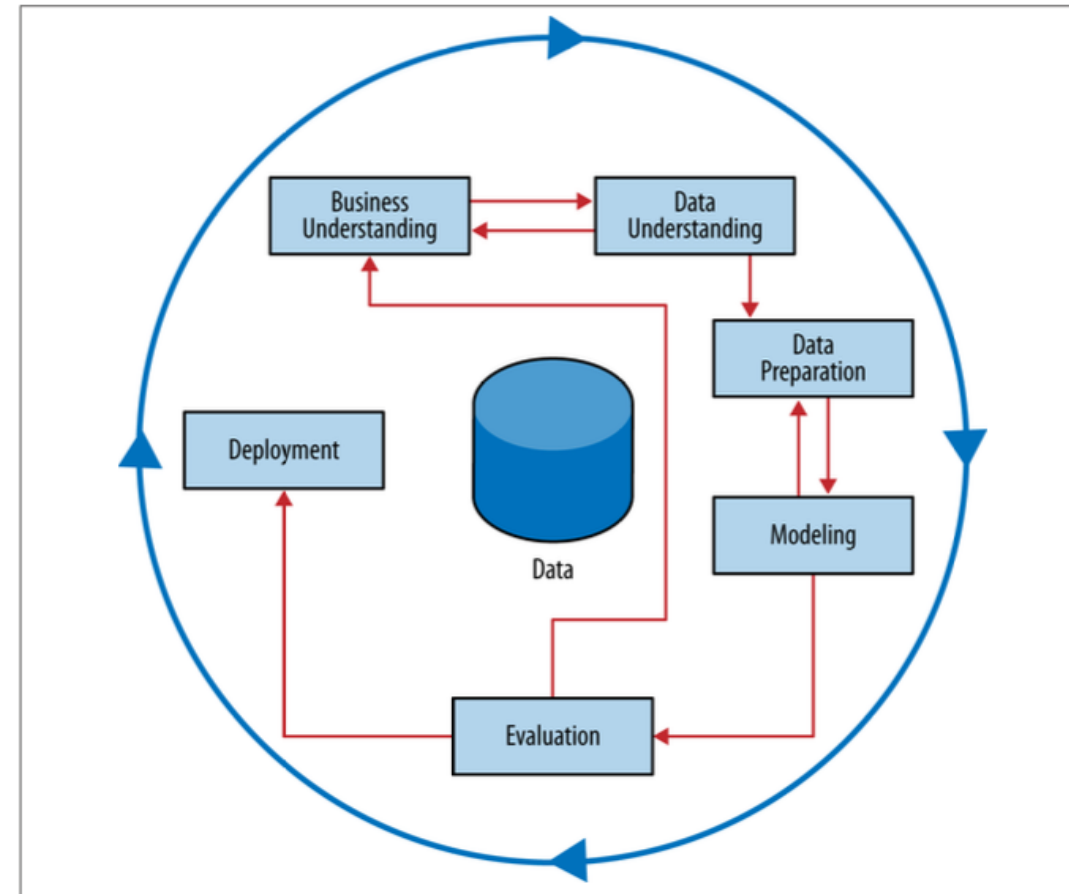


Figure 2-2. The CRISP data mining process.

Lecture
3

What is Data Preparation?

Good models need well-prepared features

Raw data is often incomplete, inconsistent, or noisy

The process of cleaning, transforming, and organizing data before modeling

Ensures that models receive reliable, usable, and valid inputs

Prevents errors, overfitting, and bias in analysis

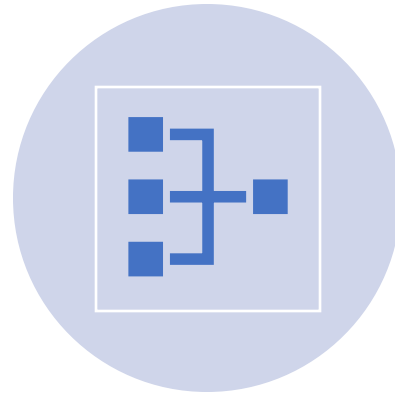
Often consumes 70–80% of the total project time

Lecture
3

Data Preparation Matters!



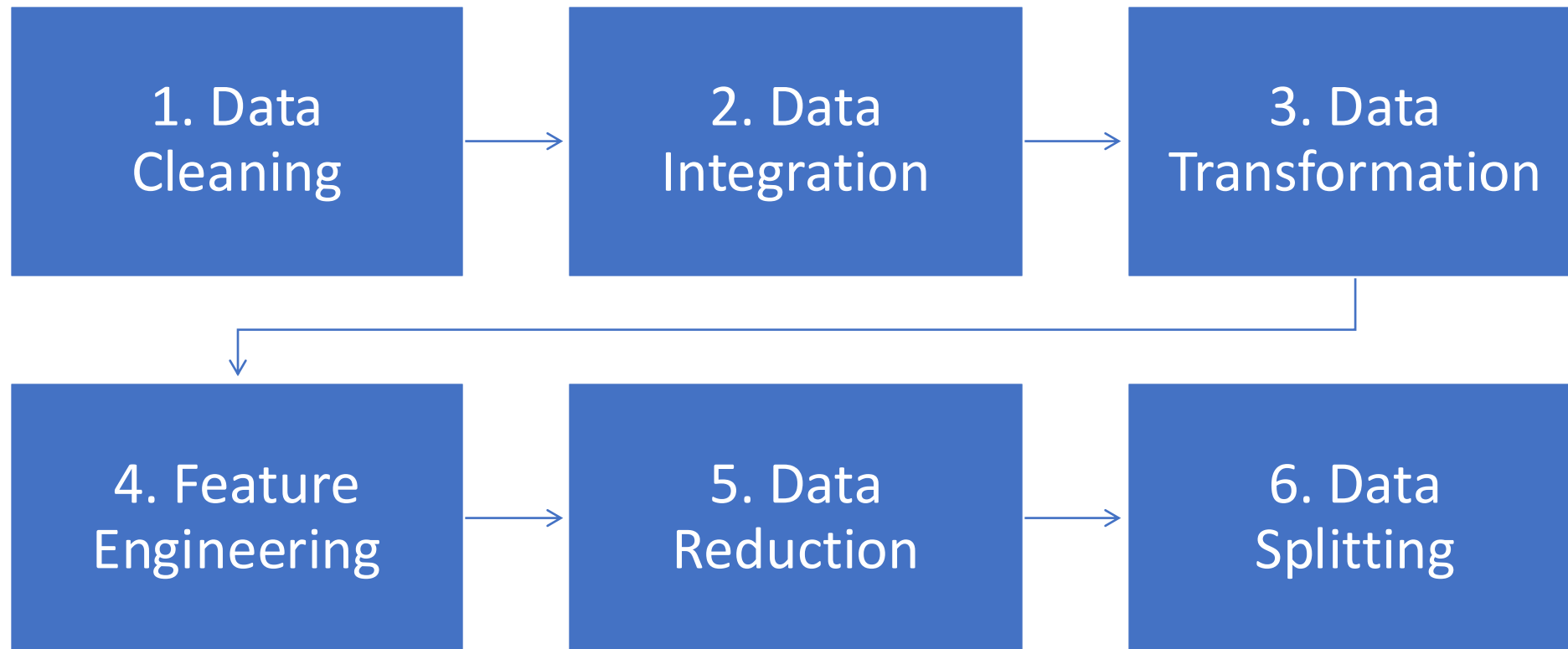
CLEANS AND
STRUCTURES RAW DATA



INCLUDES
INTEGRATION,
TRANSFORMATION,
AND ENRICHMENT



BUILDS THE
FOUNDATION FOR
ACCURATE MODELS

**Lecture
3****Main Steps in Data Preparation**

Lecture
3

Data Cleaning

Handling Missing Values

- Techniques: removal, mean/median imputation, predictive filling
- Consider: why is the data missing (random vs. systematic)?

Correcting Inconsistencies

- Standardizing formats (e.g., dates, text casing, currencies)
- Resolving conflicting entries (e.g., “NY” vs. “New York”)
- Removing Duplicates

Fixing Invalid Entries

- Out-of-range values (e.g., age = 220)
- Typos or malformed inputs (e.g., email = “abc@com”)

Outlier Detection & Treatment

- Use statistical methods (IQR, Z-score) or domain knowledge
- Decide whether to keep, transform, or remove

Lecture
3

Data Integration

- Combining data from multiple sources to produce a unified, consistent view that supports analysis, reporting, and modeling.
- It enables organizations to consolidate fragmented information, gain richer insights, and support cross-functional analysis
- Key Integration Tasks:

Match fields across
systems (schema
alignment)

Key Matching &
Joining

Data Deduplication

Resolving Conflicts

Lecture
3

Data Transformation

- Converting raw data into a suitable format for analysis and machine learning. Involves adjusting variables' structure, scale, type, and representation to improve model performance and consistency.
- Common Transformation Tasks:

Normalize or
scale data

Convert types
(e.g., date to
age)

Encode
categorical
variables

Binning /
Bucketing

Text Cleaning
and Tokenization
(for NLP)

Transformation turns *available data* into *actionable input*.

Lecture
3

Feature Engineering

- Creating, modifying, or selecting variables that improve a model's ability to learn meaningful patterns.
- Key Techniques:

Derived features:

Create new variables from existing ones

Example: Age = Today's Date – Date of Birth

Interaction features:

Combine two or more features to capture relationships

Example: Income × Education Level

Time-based features

Extract from timestamps: hour, day, time since last event

Example: Days Since Last Purchase

Text Features (for NLP):

Word counts, keyword flags, sentiment scores

Example: Review Length, Presence of Negative Words

Lecture
3

Data Reduction

- Minimizing the volume and dimensionality of data while preserving its core informational content. It makes datasets easier to process, store, and model without compromising accuracy.
- Balances model performance and interpretability
- Techniques:

Remove irrelevant or
redundant features

Dimensionality
reduction (e.g., PCA
Principal Component
Analysis))

Sampling and
Aggregation

Lecture
3

Train/Test Split

- Split data into:
 - Training set: used to fit the model
 - Testing set: used to evaluate performance
- Sometimes include validation set for tuning
- Ensures honest performance evaluation and prevents overfitting

Type of Split	Common Ratios	When to Use
Train/Test Only	70/30 or 80/20	Simple evaluations
Train/Validation/Test	60/20/20 or 70/15/15	Hyperparameter tuning required

**Lecture
3**

Sources of Data

Internal Systems

- CRM, ERP, web logs, databases

External Data

- Public datasets, APIs, social media, sensors

Manual Collection

- Surveys, forms, customer feedback

Streaming Sources

- IoT devices, clickstream, mobile apps

Data source reliability impacts preparation needs

**Lecture
3****Loading Data**

- **Getting Raw Data Into Your Environment**

Flat files:

- CSV, Excel, JSON, XML

Databases:

- SQL (MySQL, PostgreSQL, Oracle)
- NoSQL (MongoDB, Cassandra)

APIs:

- REST, GraphQL for external sources

Data Pipelines:

- Airflow, Spark, Azure Data Factory

Loading is the first technical step in preparing data

Lecture
3

Storing Data for Preparation

- Choosing Where and How to Store It

Local storage (CSV, Excel): good for small-scale work

Cloud storage (AWS S3, Azure Blob, Google Cloud): scalable and accessible

Databases and warehouses

- PostgreSQL, Snowflake, BigQuery
- Allow structured querying and indexing

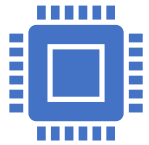
Data lakes

- Store unstructured and semi-structured data (e.g., Hadoop, Delta Lake)

Storage choice impacts speed, collaboration, and security

Lecture 3

Tools for Data Preparation



Excel (quick
fixes)



Pandas (Python)
– powerful
tabular
processing



R (tidyverse) –
statistical
cleaning and
reshaping



SQL – for
querying and
joining data



ETL tools – for
enterprise data
pipelines

Lecture
3

Common Pitfalls

Data leakage: using future info to train model

Over-imputation: filling missing data without logic

Under-documentation: losing track of changes made

**Lecture
3****Best Practices**

Document every cleaning and transformation step



Use visualizations to detect problems



Collaborate with domain experts for feature creation



Automate repeatable steps using pipelines

Lecture
3

Ready for Modelling?

- With high-quality, structured data, you can now move on to:
 - Feature Selection
 - Model Training
 - Evaluation
 - Deployment

“Better data beats fancier algorithms.” – Peter Norvig

Lecture
03

Modelling: Model Fitting and Regression



**Lecture
3****Model Fitting**

Defining the Core Task of Predictive Modeling

Model fitting means finding a function that maps inputs (X) to outputs (Y) using data

The model adjusts its internal parameters to minimize error

Example: In linear regression, this means finding the best-fitting line

Model fitting is the foundation of machine learning — it's how models learn

Lecture
3

Why do we Fit Models?

- **For the Purpose of Induction**
- Fitting turns raw data into generalizable rules
- We use models to:

Predict
unknown
outcomes

Support
business
decisions

Learn patterns
in data

Remember: The goal isn't just to match past data — it's to make future predictions

Lecture
3

What Happens During Fitting?

- Turn Input Data to Model Parameters
- Training data contains examples with known outcomes
- Model uses optimization to find the best parameters
- This process is known as induction

- Example:

Predicting price from square footage → model finds best slope/intercept

Lecture
3

Objectives of Model Induction

- What Are We Trying to Achieve?

Minimize
prediction error

Balance bias and
variance

Capture relevant
patterns (not
noise)

- Build a model that is:
 - Accurate
 - Interpretable
 - Deployable

The objective isn't just to fit — it's to fit wisely and usefully

Lecture
3

What is Generalization?

- **Generalization = model's ability to perform well on unseen data**
- The true test is not how well it fits training data, but how well it predicts future outcomes
- A good model:
 - Learns patterns, not noise
 - Performs consistently across samples

If a model can't generalize, it can't be trusted

Lecture
3

The Unknown True Function

- What We're Really Trying to Learn
- In data science, there exists a true underlying function
 - It maps inputs (X) to outputs (Y)
 - But we can never observe it directly
- Instead, we observe samples from it through noisy, imperfect data

Modeling is about approximating the truth, not finding it exactly

Lecture
3

The Model Function

- Our Best Approximation of Reality
- The model function is what we build using data
- It tries to imitate the target function as closely as possible
- Trained using:
 - Labeled data
 - Optimization (e.g., minimizing loss)

All models are approximations — but some are useful

Lecture
3

How We Approximate the Target Function

- Through the Process of Induction:
 - Learn from examples (input/output pairs)
 - Use algorithms to fit a pattern (model function)
 - Apply to new data to make predictions
- Key factors:
 - Quality & quantity of training data
 - Chosen model type
 - Optimization method

Lecture
3

What is Regression?

- Regression is a method to predict a numeric or categorical outcome
- Maps input features (X) to an output (Y)
- Helps make data-driven predictions or classifications

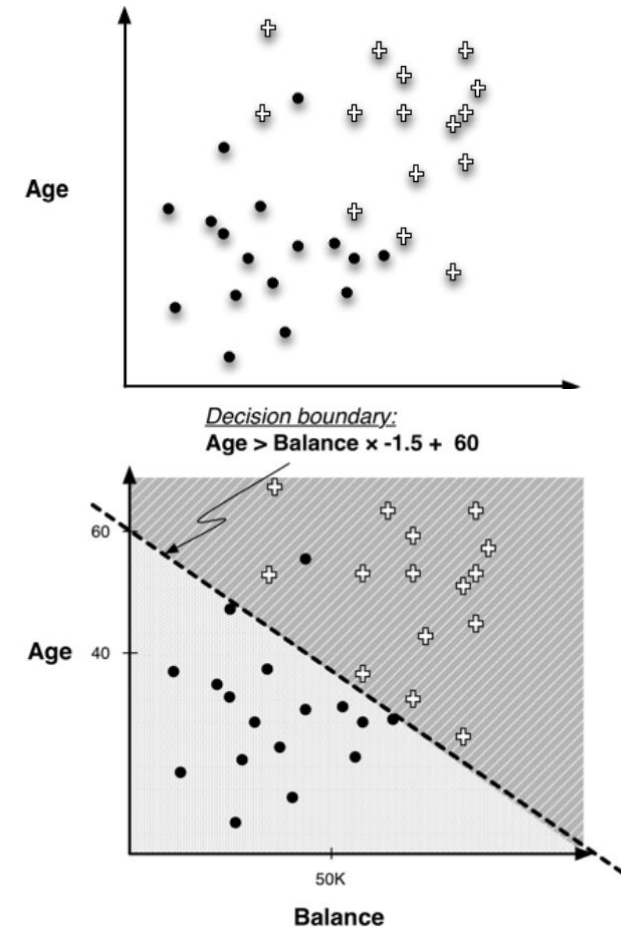
Types of Regression Models

Linear
RegressionLogistic
RegressionSupport Vector
Machines
(SVM)

Lecture
3

Linear Regression

- Predicts a continuous numerical value
- Fits a straight line ($Y = aX + b$) through data
- Goal: Minimize the error between predicted and actual values
- Example: Predicting house price from square footage
- $Y = b_0 + b_1 \cdot \text{Size}$



Simple, fast, and interpretable — ideal for linear numeric relationships

Lecture
3

Linear Regression Key Concepts and Limitations

- Assumes linearity between features and target
- Evaluation metrics:
 - **Mean Squared Error (MSE):** The average squared difference between predicted and actual values. Penalizes larger errors more than smaller ones. Lower MSE = better model performance
 - **R^2 (coefficient of determination):** The proportion of variance in the target explained by the model. Compares model against a naive baseline (mean prediction). $R^2 = 1 \rightarrow$ Perfect fit

Lecture
3

Linear Regression Key Concepts and Limitations

- Use Cases:
 - Predicting revenue, sales, prices, or demand
 - Estimating project timelines or costs
- Not suitable for:

Binary outcomes

Nonlinear
relationships

Highly correlated
features
(multicollinearity)

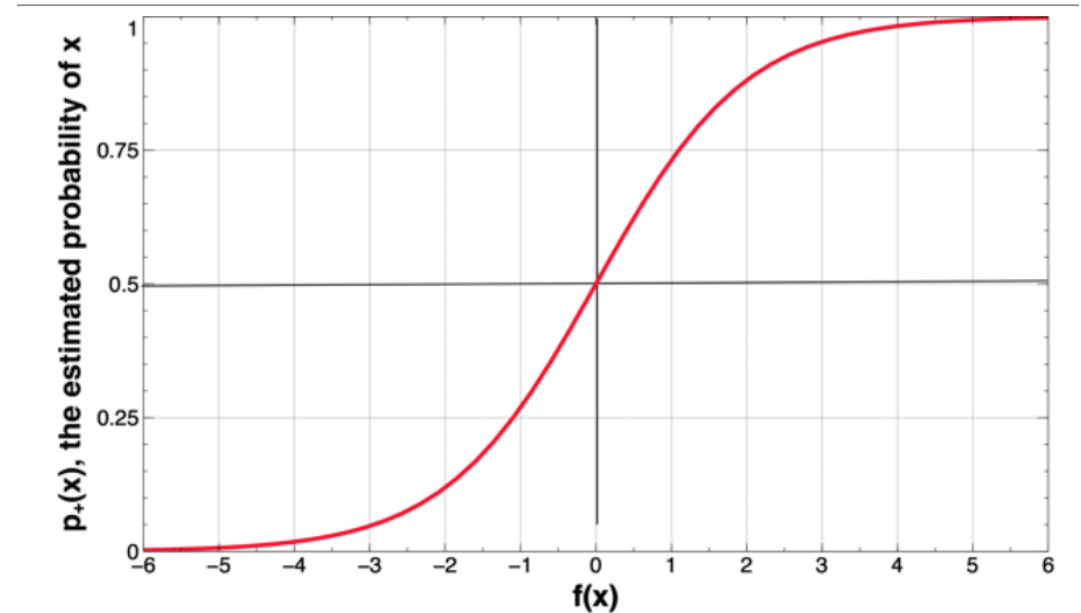
Lecture
3

Logistic Regression

- Used for binary or categorical classification
- Predicts the probability of a class
- Output is passed through a logistic (sigmoid) function
- **Sigmoid function:**

$$P(Y=1) = 1 / (1 + e^{-(b_0 + b_1X_1 + b_2X_2 + \dots)})$$

Output ranges from 0 to 1 →
interpreted as probability



Lecture
3

Logistic Regression Strengths and Limitation

- Strengths:
 - Interpretable coefficients
 - Simple and efficient for linearly separable classes
 - Works well for probability-based decision-making
- Struggles with:
 - Non-linear boundaries
 - Highly imbalanced data
 - Complex interactions between features

Lecture
3

Support Vector Machines (SVM)

- SVM is a powerful classification and regression technique
- Finds the optimal hyperplane that separates classes
- Maximizes the margin between data points and the boundary
- Works in high-dimensional spaces (e.g., text, images)
- Can create linear or nonlinear decision boundaries using kernels

SVM focuses on **robust generalization**, not just best training fit

Lecture
3

Support Vector Machines (SVM)

– Key Terms & Use Cases

- Support Vectors: Critical data points closest to the boundary
- Hyperplane: The decision boundary
- Kernel: Transforms data for nonlinear separation
- Use Cases:
 - Image recognition
 - Bioinformatics (gene classification)
 - Text classification and sentiment analysis

Lecture 3

Strengths and Limitations of SVM

Strengths	Limitations
Powerful for Complex Classification • Works well with nonlinear boundaries using kernel tricks • Handles high-dimensional spaces (e.g., text, images) • Less prone to overfitting with proper regularization • Effective even with small datasets Ideal for tasks where linear models fail and interpretability is less important	Computationally intensive on large datasets Hard to interpret compared to regression models Sensitive to: • Choice of kernel • Regularization parameter (C) • Feature scaling

Best suited when performance matters more than explanation

Lecture 3

Types of Regression Comparison

Model Type	Output Type	Interpretability	Key Feature	Suitable for	Common Use Cases
Linear Regression	Continuous (numeric)	High	Line Fitting	Trend/pattern prediction	Sales forecasting, pricing
Logistic Regression	Binary/Categorical/probability	Moderate	Binary Classification	Yes/No Outcomes	Churn prediction, fraud detection
SVM	Binary or multiclass (class label)	Low	Margin Optimization	Complex/High-Dimensional data	Classification, margin maximization

**Lecture
3**

Choosing the Right Model

- Not all regression techniques are suited for every task
- Selecting the right model improves accuracy, interpretability, and business impact

Use Linear Regression when:

Your target is a continuous variable

Relationships appear linear

Use Logistic Regression when:

You need probabilities or binary classification

Interpretability is important

Use SVM when:

Data has complex, nonlinear patterns

Accuracy and margin-based separation are key

Lecture
3

Model Assumptions

- Every Model Makes Assumptions
- Examples:
 - Linear regression assumes linear relationships
 - Logistic regression assumes log-odds linearity
 - SVM assumes margin-based separability
- Violating these assumptions can reduce accuracy and Lead to misleading conclusions

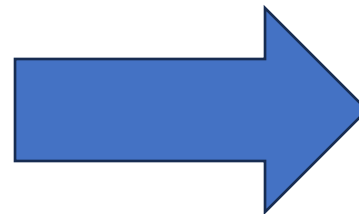
Understanding assumptions helps you choose the right model for the right data

**Lecture
3****Examples****Forecasting Revenue Based
on Ad Spend**

Goal: Predict **monthly sales**
based on marketing investment

Inputs: TV, radio, online ad
budget

Output: Predicted sales
(numeric)

**Linear Regression:**

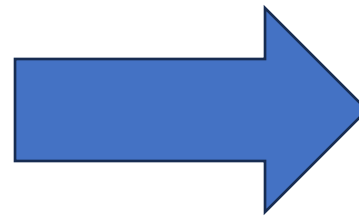
- ✓ Relationship is mostly linear
- ✓ Output is continuous
- ✓ Quick and interpretable for marketing team

**Lecture
3****Examples****Detecting Handwritten Digits**

Goal: Classify handwritten digits
(0–9)

Inputs: Pixel values from images

Output: Predicted digit class

**SVM:**

- ✓ Data is high-dimensional
- ✓ Decision boundary is nonlinear
- ✓ Accuracy is more important than interpretability

Lecture
3

Key Points

Data Preparation Is Essential

Clean, integrated, and well-structured data leads to better models.

Model Fitting = Learning Patterns

Fitting means training models to minimize prediction error.

Choose the Right Regression Tool

Linear = numeric prediction, Logistic = classification, SVM = complex boundaries

Evaluate with the Right Metrics

Use **MSE** for error measurement, **R²** for model explanation.

Train/Test Splits Ensure Reliability

Splitting data detects overfitting and improves generalization.

For Discussion Panel and Next Lecture:

- Use the open-source Telco Customer Churn dataset available on [Kaggle](#).
- Apply Linear Regression to Forecast Customer Lifetime Value (CLV)

Lecture 3

For Discussion Panel and Next Lecture





Thank You