DSA 610 Redesign, Lecture 12 Outline

Lecture Outline: Data Retention and Policies
Duration: 50 minutes

---

1. Introduction to Data Retention (5 minutes)

- **Objective:** Understand the concept of data retention, its importance, and the general principles guiding it.
- **Content:**
  - **Definition:** Data retention refers to the policies and practices of storing data for a specific period to meet legal, regulatory, or business requirements.
  - **Purpose:** Ensures data availability for future reference, compliance, and decision-making.

---

2. Data Retention Regulations (15 minutes)

a. General Data Protection Regulation (GDPR)

- **Overview:** A regulation in EU law on data protection and privacy for individuals.
- **Key Points:**
  - **Data Minimization:** Collect only the data necessary for the intended purpose.
  - **Retention Period:** Data should be kept only as long as necessary for the processing purpose.
  - **Right to Erasure:** Individuals can request deletion of their data under certain conditions.

b. Health Insurance Portability and Accountability Act (HIPAA)

- **Overview:** U.S. law that mandates data privacy and security provisions for safeguarding medical information.
- **Key Points:**
  - **Retention Period:** Healthcare providers must retain patient records for a minimum of 6 years.
  - **Security Requirements:** Ensures confidentiality and security of health data.

c. Sarbanes-Oxley Act (SOX)

- **Overview:** U.S. law that mandates financial record-keeping and reporting.
- **Key Points:**
  - **Retention Period:** Requires retention of financial records and documents for at least 7 years.
  - **Compliance:** Aims to prevent corporate fraud and ensure accuracy of financial statements.

d. Federal Information Security Management Act (FISMA)

- **Overview:** U.S. law requiring federal agencies to secure information systems.
- **Key Points:**
  - **Retention Policies:** Agencies must implement policies to protect data and maintain records for audit and compliance purposes.

e. Payment Card Industry Data Security Standard (PCI DSS)

- **Overview:** Industry standard for securing credit card information.
- **Key Points:**
  - **Retention Period:** Cardholder data must be encrypted and retained only as long as necessary for business purposes.

---

3. Backup vs. Archiving (10 minutes)

a. Backup

- **Definition:** Process of creating copies of data to protect against loss or corruption.
- **Purpose:** To restore data in case of system failure, data corruption, or accidental deletion.
- **Characteristics:**
    - **Frequency:** Regularly scheduled (e.g., daily, weekly).
    - **Retention:** Short-term, often kept for a limited period.
    - **Example:** Daily incremental backups, system state backups.

b. Archiving

- **Definition:** Storing data that is no longer actively used but needs to be retained for long-term purposes.
- **Purpose:** To keep historical data accessible and compliant with legal or business requirements.
- **Characteristics:**
    - **Frequency:** Occasional, often based on data lifecycle.
    - **Retention:** Long-term, sometimes indefinitely.
    - **Example:** Archived emails, old financial records.

c. Differences

- **Purpose:** Backups for immediate recovery; Archiving for long-term storage and compliance.
- **Frequency:** Backups are frequent; Archiving is less frequent and usually done when data is no longer actively used.
- **Accessibility:** Backup data is readily accessible for recovery; Archived data may be less accessible and stored in a more cost-effective manner.

---

4. Data Retention Policies (15 minutes)

a. Importance of Data Retention

- **Compliance:** Ensures adherence to legal and regulatory requirements.
- **Business Continuity:** Supports ongoing operations and decision-making.
- **Historical Records:** Provides access to historical data for analysis, auditing, and reference.

b. Reasons for Retaining Data

- **Legal Requirements:** Comply with laws and regulations that mandate data retention.
- **Operational Needs:** Maintain data for business operations, customer support, and analysis.
- **Historical Analysis:** Access to historical data for trend analysis, reporting, and strategic planning.

c. Reasons for Not Retaining Everything

- **Cost:** Storing large volumes of data can be expensive and resource-intensive.
- **Data Privacy:** Reduces risk of exposing sensitive information and potential breaches.
- **Compliance:** Avoids non-compliance with regulations that require data minimization and controlled access.

d. Developing a Data Retention Policy

- **Steps:**
    - **Identify Data Types:** Classify data based on its importance, sensitivity, and regulatory requirements.
    - **Define Retention Periods:** Specify how long each type of data should be retained.
    - **Implement Procedures:** Establish processes for data backup, archiving, and disposal.
    - **Review and Update:** Regularly review policies to ensure they remain current with regulations and business needs.

---

5. Q&A and Discussion (5 minutes)

- **Objective:** Address questions and discuss practical considerations related to data retention.
- **Content:**

- **Q&A Session:** Open the floor for student questions.
- **Discussion:** Explore real-world scenarios and challenges in implementing data retention policies.

Key Takeaways

- **Data Retention:** Crucial for compliance, business continuity, and historical analysis.
- **Regulations:** GDPR, HIPAA, SOX, FISMA, and PCI DSS provide guidelines on data retention.
- **Backup vs. Archiving:** Understand the differences to manage data effectively.
- **Policies:** Develop and maintain data retention policies to balance compliance, cost, and operational needs.

Resources:
What is data retention?: https://bigid.com/blog/what-is-data-retention/
Guidelines for developing a data retention policy: https://www.ispartnersllc.com/blog/standards-developing-data-retention-policy/
Best Practices and Template: https://www.ispartnersllc.com/blog/standards-developing-data-retention-policy/
Data Retention and GDPR: https://www.dpocentre.com/data-retention-and-the-gdpr-best-practices-for-compliance/
Data Retention HIPAA: https://www.hipaajournal.com/hipaa-retention-requirements/
Sarbannes-Oxley: https://www.armstrongarchives.com/sox-data-retention-requirements/
FISMA Retention policies: https://www.opusguard.com/kb/data-retention-laws-and-regulations/
PCI DSS policies:
https://www.treasury.uillinois.edu/merchant_card_services/data_security/System_pcidss_policies
Backing up Data: https://cloudian.com/guides/data-backup/data-backup-in-depth/
Data Archiving: https://www.druva.com/glossary/what-is-data-archiving-definition-and-related-faqs
Backup vs. Archiving: https://www.seagate.com/blog/backup-vs-archiving-the-key-differences-between-the-both/
Data Retention Policies: https://www.techtarget.com/searchdatabackup/definition/data-retention-policy
https://www.proofpoint.com/us/threat-reference/data-retention-policy
Data Retention Policy Sample: https://www.odga.virginia.gov/media/governorvirginiagov/chief-data-officer/pdf/COV-Data-Retention-Policy-Template.pdf
Buffalo State University Data Retention Policy:
https://financeandmanagement.buffalostate.edu/retention-and-disposition

Lecture Outline: Communicating Results in Data Analysis
Duration: 50 minutes

---

1. Introduction to Communicating Results (5 minutes)

- **Objective:** Understand the importance of effectively communicating data analysis results to various audiences.
- **Content:**
  - **Definition:** Communicating results involves presenting findings in a clear, engaging, and understandable manner.
  - **Purpose:** To ensure that insights are actionable and comprehensible to stakeholders.

---

2. Storytelling and Narrative (15 minutes)
a. The Role of Storytelling in Data Analysis

- **Definition:** Using a narrative to present data findings in a way that is engaging and memorable.
- **Purpose:** To provide context, highlight key insights, and make data more relatable.
- **Components:**
    - **Introduction:** Set the stage with background information.
    - **Challenge/Problem:** Describe the issue or question being addressed.
    - **Solution/Findings:** Present the analysis and insights.
    - **Conclusion/Call to Action:** Summarize findings and suggest next steps.

b. Crafting a Data Story

- **Identify Your Audience:** Tailor the story to their level of understanding and interests.
- **Structure Your Narrative:** Use a clear structure to guide the audience through the data.
- **Use Examples:** Provide concrete examples to illustrate points.
- **Visualize Key Insights:** Use charts and graphs to support the narrative.

---

3. Good Graphs vs. Bad Graphs (10 minutes)

a. Characteristics of Good Graphs

- **Clarity:** Easy to understand at a glance.
- **Relevance:** Directly related to the key message or insight.
- **Simplicity:** Avoid clutter and focus on essential information.
- **Accuracy:** Represent data truthfully without distortion.

b. Examples of Good Graphs

- **Line Graphs:** Show trends over time.
- **Bar Charts:** Compare quantities across categories.
- **Pie Charts:** Illustrate proportions of a whole.
- **Histograms:** Display distribution of data.

c. Characteristics of Bad Graphs

- **Overly Complex:** Too much information or clutter.
- **Misleading:** Distort or misrepresent data.
- **Inappropriate Choices:** Use of unsuitable graph types for the data.
- **Poor Design:** Inconsistent scales, missing labels, or ambiguous legends.

d. Examples of Bad Graphs

- **3D Charts:** Can be misleading and hard to interpret.
- **Pie Charts with Too Many Segments:** Difficult to compare slices.
- **Graphs with Unclear Labels or Scales:** Confuse the audience.

---

4. Communicating with Different Audiences (10 minutes)

a. Communicating with Experts

- **Focus:** Detailed analysis, methodologies, and technical aspects.
- **Content:** Use technical terminology and in-depth data insights.
- **Approach:** Present comprehensive results, statistical significance, and model details.

b. Communicating with Lay Audience

- **Focus:** High-level insights, implications, and actionable recommendations.
- **Content:** Use simple language and avoid technical jargon.
- **Approach:** Use clear visuals, analogies, and concise explanations.

c. Adapting Your Communication

- **Tailor Content:** Adjust complexity based on the audience's familiarity with the subject.
- **Use Analogies:** Simplify complex concepts with relatable examples.
- **Focus on Impact:** Emphasize how the findings affect the audience or their decisions.

---

5. Common Errors to Avoid (5 minutes)

a. Overloading with Information

- **Issue:** Presenting too much data at once can overwhelm the audience.
- **Solution:** Focus on key insights and present supporting details as needed.

b. Neglecting Visualization Best Practices

- **Issue:** Using misleading or poorly designed graphs.
- **Solution:** Follow best practices for graph design and accuracy.

c. Ignoring Audience Needs

- **Issue:** Failing to consider the audience's background and interests.
- **Solution:** Tailor your communication to the audience's level of understanding and priorities.

d. Overgeneralizing or Misrepresenting Data

- **Issue:** Making broad claims without sufficient evidence.
- **Solution:** Base conclusions on solid data and clearly state any limitations.

---

6. Q&A and Discussion (5 minutes)

- **Objective:** Address questions and discuss practical considerations for effective communication.
- **Content:**
    - **Q&A Session:** Open the floor for student questions.
    - **Discussion:** Explore real-world scenarios and examples of effective and ineffective communication.

Key Takeaways

- **Storytelling:** Use narratives to make data engaging and meaningful.
- **Good vs. Bad Graphs:** Understand what makes effective visualizations and avoid common pitfalls.
- **Audience Adaptation:** Tailor communication based on the audience's expertise and needs.
- **Avoid Common Errors:** Focus on clarity, relevance, and accuracy in presenting data.

Resources:

Preventing Data Overload: https://nexightgroup.com/preventing-data-overload-communicating-data-analysis-effectively/

Data Analysis and Communication: https://www.urban.org/measure4change-performance-measurement-playbook/data-analysis-and-communication

How to Communicate Results to Business Stakeholders: https://www.pragmaticinstitute.com/resources/articles/data/comprehensive-guide-how-to-communicate-data-insights-to-business-stakeholders/

Data Scientist's Guide to Communicating Results: https://www.linkedin.com/pulse/data-scientists-guide-communicating-results-data-semantics/

8 Examples of Storytelling with Data: https://shorthand.com/the-craft/examples-of-powerful-data-storytelling/index.html

Data Storytelling with Examples: https://www.microsoft.com/en-us/power-platform/products/power-bi/topics/data-storytelling

How to tell a great story with Data: https://www.thoughtspot.com/data-trends/best-practices/data-storytelling

Good and Bad Examples of Data Visualizations: https://www.polymersearch.com/blog/10-good-and-bad-examples-of-data-visualization

Good and Bad Graphs: https://www.stat.auckland.ac.nz/~ihaka/120/Lectures/lecture03.pdf

Adapting Communication Styles: https://www.leadershipsuccess.co/effective-communication/adapting-to-the-person
Audience Adaptation: https://www.comm.pitt.edu/audience-adaptation
9 Common Errors: https://dashthis.com/blog/mistakes-in-data-analysis/

Lecture Outline: Ensemble Methods in Python
Duration: 50 minutes

---

1. Introduction to Ensemble Methods (5 minutes)

- **Objective:** Understand the concept and advantages of ensemble methods in machine learning.
- **Content:**
  - **Definition:** Ensemble methods combine multiple models to improve prediction performance and robustness.
  - **Purpose:** To leverage the strengths of different models and mitigate their weaknesses, leading to better generalization.

---

2. General Overview of Ensemble Methods (10 minutes)

a. Types of Ensemble Methods

- **Bagging (Bootstrap Aggregating):** Reduces variance by training multiple models on different subsets of the data and averaging their predictions.
- **Boosting:** Reduces bias by sequentially training models where each new model corrects the errors of the previous one.
- **Stacking (Stacked Generalization):** Combines multiple models by training a meta-model to learn how to best combine their predictions.

b. Key Concepts

- **Base Learners:** Individual models used in the ensemble.
- **Meta-Learner:** A model used in stacking to combine the predictions of base learners.
- **Aggregation:** The process of combining predictions from multiple models.

---

3. Bagging in Python (10 minutes)

a. Overview

- **Definition:** Bagging involves training multiple instances of the same model on different bootstrapped subsets of the data and averaging their predictions.

b. Example: Bagging with Decision Trees

- **Model:** BaggingClassifier from sklearn using DecisionTreeClassifier as the base learner.

```
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.ensemble import BaggingClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score

# Load dataset
data = load_iris()
X = data.data
y = data.target

# Split data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
```

```
# Initialize BaggingClassifier
bagging_clf = BaggingClassifier(base_estimator=DecisionTreeClassifier(), n_estimators=50,
random_state=42)

# Train model
bagging_clf.fit(X_train, y_train)

# Predict and evaluate
y_pred = bagging_clf.predict(X_test)
print("Bagging Accuracy:", accuracy_score(y_test, y_pred))
```

4. Boosting in Python (15 minutes)

a. Overview

- **Definition:** Boosting sequentially trains models, with each model focusing on the errors of the previous one, and combines their predictions.

b. Example: AdaBoost with Decision Trees

- **Model:** AdaBoostClassifier from sklearn using DecisionTreeClassifier as the base learner.

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.tree import DecisionTreeClassifier

# Initialize AdaBoostClassifier
ada_clf = AdaBoostClassifier(base_estimator=DecisionTreeClassifier(max_depth=1), n_estimators=50,
random_state=42)

# Train model
ada_clf.fit(X_train, y_train)

# Predict and evaluate
y_pred = ada_clf.predict(X_test)
print("AdaBoost Accuracy:", accuracy_score(y_test, y_pred))
```

c. Other Boosting Methods

- **Gradient Boosting:** Trains models in a stage-wise manner, optimizing for the residual errors of previous models.
- **XGBoost:** An optimized version of gradient boosting, offering better performance and scalability.

---

5. Stacking in Python (10 minutes)

a. Overview

- **Definition:** Stacking combines predictions from multiple models (base learners) using a meta-model that learns how to best combine these predictions.

b. Example: Stacking with Logistic Regression as Meta-Learner

- **Models:** LogisticRegression as the meta-learner, DecisionTreeClassifier and KNeighborsClassifier as base learners.

```
from sklearn.ensemble import StackingClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import KNeighborsClassifier
```

```
# Initialize base learners
base_learners = [
   ('decision_tree', DecisionTreeClassifier()),
   ('knn', KNeighborsClassifier())
]

# Initialize StackingClassifier
stacking_clf = StackingClassifier(estimators=base_learners, final_estimator=LogisticRegression(), cv=5)

# Train model
stacking_clf.fit(X_train, y_train)

# Predict and evaluate
y_pred = stacking_clf.predict(X_test)
print("Stacking Accuracy:", accuracy_score(y_test, y_pred))
```

6. Key Considerations and Best Practices (5 minutes)

a. Choosing Base Learners

- **Diversity:** Use diverse models to capture different aspects of the data.
- **Complexity:** Ensure that base learners are not too complex or too simple.

b. Avoiding Overfitting

- **Validation:** Use cross-validation to evaluate ensemble performance.
- **Regularization:** Regularize base models to avoid overfitting.

c. Computational Efficiency

- **Training Time:** Ensembles can be computationally intensive. Consider using parallel processing or optimizing base models.

---

7. Q&A and Discussion (5 minutes)

- **Objective:** Address questions and discuss practical considerations for using ensemble methods.
- **Content:**
    - **Q&A Session:** Open the floor for student questions.
    - **Discussion:** Explore real-world scenarios where ensemble methods are effective and common challenges faced.

Key Takeaways

- **Ensemble Methods:** Enhance model performance by combining multiple models through bagging, boosting, and stacking.
- **Python Implementation:** Utilize sklearn for practical examples of ensemble methods.
- **Considerations:** Choose diverse base learners, manage overfitting, and optimize computational efficiency.

Resources:

Ensemble Methods in Python: https://www.geeksforgeeks.org/ensemble-methods-in-python/
Sci-kit Learn: Ensemble Methods: https://scikit-learn.org/stable/modules/ensemble.html
Ensemble Modeling Tutorial: https://www.datacamp.com/tutorial/ensemble-learning-python
Bagging: https://www.w3schools.com/python/python_ml_bagging.asp
Random Forest Regression: https://www.geeksforgeeks.org/random-forest-regression-in-python/

Boosting: https://www.datacamp.com/tutorial/what-is-boosting
Adaboost: https://www.datacamp.com/tutorial/adaboost-classifier-python
Gradient Boosting: https://www.geeksforgeeks.org/ml-gradient-boosting/
XGBoost: https://xgboost.readthedocs.io/en/stable/python/python_intro.html
Stacking: https://www.geeksforgeeks.org/stack-in-python/
Bagging vs. Boosting vs. Stacking: https://www.baeldung.com/cs/bagging-boosting-stacking-ml-ensemble-models