Digital Finance 3: Technology in Finance

Lesson 26: Financial Data for AI/ML

FHGR

December 11, 2025

Learning Objectives

By the end of this lesson, you will be able to:

- Distinguish between structured and unstructured financial data
- Identify major sources and vendors of financial data
- Understand the alternative data revolution and its applications
- Recognize data quality issues and preprocessing requirements
- Explain GDPR and privacy implications for financial ML
- Describe basic feature engineering concepts

Structured vs. Unstructured Data

Structured Data:

- Organized in tables (rows, columns)
- Predefined schema
- Easy to query (SQL)
- Numerical or categorical

Examples in Finance:

- Stock prices (OHLCV)
- Financial statements
- Credit bureau data
- Transaction records
- Market microstructure (order book)

Unstructured Data:

- No predefined format
- Free-form text, images, audio
- Requires NLP/computer vision
- 80-90% of enterprise data

Examples in Finance:

- News articles, press releases
- Earnings call transcripts
- SEC filings (10-K, 8-K)
- Social media (Twitter, Reddit)
- Analyst reports

Modern ML: Combines both types (e.g., sentiment scores from text as features in tabular models).

Traditional Financial Data Sources

Market Data:

- Stock prices (exchanges)
- Bond yields (TRACE, Bloomberg)
- Derivatives (CME, Eurex)
- FX rates (interbank, EBS)
- Crypto (Coinbase, Binance)

Fundamental Data:

- Financial statements (EDGAR, SEDAR)
- Company events (earnings, M&A)
- Economic indicators (BLS, Fed, ECB)
- Industry metrics (PMI, CPI)

Credit/Risk Data:

- Credit bureaus (Experian, Equifax, TransUnion)
- Ratings (Moody's, S&P, Fitch)
- Loan performance data
- Default histories

Major Vendors:

- Bloomberg Terminal ($20-25k/year)
- Refinitiv (formerly Thomson Reuters)
- FactSet
- S&P Capital IQ
- Morningstar

Trend: Declining costs for basic data (Yahoo Finance free), but premium data remains expensive.

The Alternative Data Revolution

What is Alternative Data?

- Non-traditional data sources
- Often unstructured or semi-structured
- Provides early signals
- Competitive edge (information advantage)

Categories:

1. **Web-scraped:** Prices, reviews, job postings
2. **Sensor/IoT:** Satellite, credit cards, mobile location
3. **Social:** Twitter sentiment, Reddit mentions
4. **Business:** Email receipts, app usage

Example Use Cases:

- Satellite images: Count cars in parking lots (retail sales proxy)
- Credit card data: Real-time consumer spending
- Job postings: Company growth indicators
- App downloads: User engagement trends
- Shipping data: Supply chain analysis

Market Size:

- $1.7B in 2020
- Projected $17B by 2027
- Hedge funds are largest buyers

Challenges: Quality control, legal/ethical concerns, data decay (alpha decay).

Alternative Data: Examples in Detail

Satellite Imagery:

- Providers: Orbital Insight, RS Metrics
- Use: Count oil tanks, construction activity
- Example: China steel production estimates
- Frequency: Daily to weekly
- Cost: $10k-100k+ per year

Credit Card Transactions:

- Providers: Facteus, Second Measure
- Use: Real-time revenue tracking
- Example: Restaurant chain performance
- Privacy: Aggregated, anonymized

Social Media Sentiment:

- Providers: RavenPack, Bloomberg sentiment
- Use: Market mood, event detection
- Example: Tweet volume predicting volatility
- Challenges: Noise, manipulation

Web Traffic:

- Providers: SimilarWeb, Alexa (discontinued)
- Use: Company engagement metrics
- Example: E-commerce site visits
- Limitation: Sample-based estimates

Key Question: Does alternative data provide genuine alpha or just noise? Evidence: Mixed, diminishing returns as adoption increases (alpha decay).

Data Quality: The GIGO Principle

Garbage In, Garbage Out:

- ML models amplify data quality issues
- No algorithm fixes bad data
- Quality ¿ Quantity (usually)

Common Data Problems:

- **Missing values:** Deletions, NaN, nulls
- **Outliers:** Errors vs. true extremes
- **Inconsistencies:** Units, formats, definitions
- **Duplicates:** Same record multiple times
- **Errors:** Typos, wrong values

Finance-Specific Issues:

- **Survivorship bias:** Only successful firms remain
- **Look-ahead bias:** Using future information
- **Corporate actions:** Splits, dividends, mergers
- **Restatements:** Accounting changes, revisions
- **Stale data:** Delayed or infrequent updates

Impact on ML:

- Biased predictions
- Overfitting to noise
- Poor generalization
- Misleading performance metrics

Best Practice: Spend 50-80% of project time on data cleaning and validation.

Data Preprocessing Pipeline

Step 1: Data Collection

- Define requirements
- Source identification
- API integration or downloads
- Compliance checks

Step 2: Cleaning

- Handle missing values (drop, impute, flag)
- Remove duplicates
- Correct errors (domain knowledge)
- Outlier treatment (winsorize, cap)

Step 3: Transformation

- Normalization/standardization
- Log transforms (skewed distributions)
- Date/time parsing
- Encoding categoricals

Step 4: Feature Engineering

- Create derived features
- Lag variables (time series)
- Interactions (cross-products)
- Domain-specific ratios

Step 5: Validation

- Statistical checks (distributions)
- Consistency tests
- Cross-field validation
- Expert review

Step 6: Versioning

- Track data lineage
- Version control for datasets
- Reproducibility

Automation: Modern ML pipelines use tools like Apache Airflow, Prefect for orchestration.

Handling Missing Data

Why Data is Missing:

1. **MCAR** (Missing Completely At Random): Pure chance, no pattern
2. **MAR** (Missing At Random): Related to observed data
3. **MNAR** (Missing Not At Random): Related to unobserved value itself

Finance Example:

- MCAR: Random system glitch
- MAR: Small firms don't report segment data
- MNAR: Firms hide bad performance

Strategies:

- **Deletion:** Drop rows/columns (only if ¡ 5% missing, MCAR)
- **Mean/Median imputation:** Replace with average (simple, biased)
- **Forward/backward fill:** Time series (assumes persistence)
- **Model-based:** Predict missing values (KNN, regression)
- **Indicator variable:** Flag missingness as feature

Best Practice:

- Understand WHY data is missing
- Test sensitivity to imputation method
- Document assumptions

Feature Engineering Fundamentals

What is Feature Engineering?

- Creating new variables from raw data
- Domain knowledge + creativity
- Often more important than algorithm choice
- "Applied feature engineering beats fancy algorithms"

Common Techniques:

- **Ratios:** P/E, Debt/Equity, ROE
- **Differences:** Price changes, growth rates
- **Lags:** Yesterday's return, 30-day moving avg
- **Aggregations:** Sum, mean, max over time window
- **Interactions:** Sector × Size, Region × Industry

Finance-Specific Features:

- Technical indicators (RSI, MACD, Bollinger Bands)
- Volatility measures (realized, implied)
- Momentum (12-month return)
- Value factors (book-to-market)
- Quality metrics (accruals, earnings quality)

Avoid:

- Leakage (using future information)
- High cardinality categoricals (too many levels)
- Perfectly correlated features (redundant)
- Features with no variance

Art + Science: Combines domain expertise with systematic experimentation.

GDPR and Data Privacy Regulations

GDPR Key Principles (EU, 2018):

- **Lawfulness:** Legal basis for processing
- **Purpose limitation:** Specific, explicit purposes
- **Data minimization:** Collect only necessary data
- **Accuracy:** Keep data up-to-date
- **Storage limitation:** Retain only as long as needed
- **Integrity/confidentiality:** Secure processing

Individual Rights:

- Right to access
- Right to erasure ("right to be forgotten")
- Right to explanation (Article 22)

Implications for ML:

- Consent requirements (explicit for sensitive data)
- Anonymization challenges (re-identification risk)
- Model explainability (if automated decision-making)
- Data retention policies
- Cross-border data transfers (adequacy decisions)

Other Regulations:

- CCPA (California Consumer Privacy Act)
- LGPD (Brazil)
- POPIA (South Africa)

Penalties: Up to 4% of global revenue or 20M EUR (whichever is higher).

Anonymization vs. Pseudonymization

Anonymization:

- Irreversible removal of identifiers
- No longer personal data under GDPR
- Techniques:
  - Aggregation
  - Noise addition (differential privacy)
  - Generalization (age → age range)
- Challenge: Re-identification risk (AOL search data, Netflix prize)

Re-identification Example: 87% of US population identifiable with just:

- ZIP code
- Gender
- Date of birth

Pseudonymization:

- Replace identifiers with pseudonyms (tokens)
- Reversible with key
- Still personal data under GDPR
- Techniques:
  - Hashing
  - Encryption
  - Tokenization
- Reduces risk but doesn't eliminate it

Finance Use Cases:

- Credit scoring: Pseudonymized customer IDs
- Fraud detection: Anonymized transaction patterns
- AML: Must balance privacy with compliance

Best Practice: Privacy by design (build privacy into system architecture from the start).

Data Bias: Types and Examples

Selection Bias:

- Sample not representative of population
- Example: Credit model trained only on approved loans (missing rejected applicants who would have repaid)

Survivorship Bias:

- Only successful entities remain in dataset
- Example: Mutual fund returns (failed funds disappear)
- Impact: Overstates historical performance

Historical Bias:

- Past discrimination baked into data
- Example: Redlining in mortgage data
- Model learns and perpetuates bias

Measurement Bias:

- Systematic errors in data collection
- Example: Self-reported income (underreporting)

Temporal Bias:

- Data from one time period doesn't generalize
- Example: Pre-2008 credit models failed post-crisis

Mitigation Strategies:

- Diverse, representative datasets
- Bias audits (fairness metrics)
- Reweighting samples
- Adversarial debiasing
- Domain expert review

Key Insight: Bias in data leads to biased models, which lead to unfair outcomes. Proactive detection is essential.

Time Series Data: Special Considerations

Unique Challenges:

- **Temporal dependence:** Observations not independent
- **Non-stationarity:** Statistical properties change over time
- **Seasonality:** Recurring patterns (quarterly earnings)
- **Trends:** Long-term drift
- **Structural breaks:** Regime changes (crises)

Cannot Use Standard ML:

- Random train/test split violates temporal order
- Cross-validation needs time-aware folds
- Risk of look-ahead bias

Proper Approach:

- **Walk-forward validation:** Train on past, test on future
- **Expanding window:** Grow training set over time
- **Rolling window:** Fixed-size window (adapts to recent data)

Feature Engineering:

- Lags (t-1, t-2, ..., t-n)
- Rolling statistics (moving averages, volatility)
- Date/time features (day of week, month, quarter)
- Event indicators (earnings announcement, FOMC)

Stationarity Tests:

- Augmented Dickey-Fuller (ADF)
- Differencing if non-stationary

Data Storage and Infrastructure

Storage Options:

- **Relational databases:** PostgreSQL, MySQL (structured data, ACID transactions)
- **NoSQL:** MongoDB, Cassandra (unstructured, scale)
- **Data warehouses:** Snowflake, Redshift (analytics)
- **Data lakes:** S3, Azure Data Lake (raw data, all formats)
- **Time series DB:** InfluxDB, TimescaleDB (high-frequency data)

File Formats:

- CSV (simple, inefficient)
- Parquet (columnar, compressed, fast)
- HDF5 (hierarchical, scientific)

Cloud vs. On-Premise:

- Cloud: Scalability, cost-effective (pay-as-you-go)
- On-prem: Control, security (for sensitive data)
- Hybrid: Regulatory compliance + flexibility

Data Governance:

- Data catalog (metadata management)
- Access control (role-based)
- Audit logs (who accessed what, when)
- Data quality monitoring
- Lineage tracking (source to destination)

Cost Considerations:

- Storage: Cheap (S3 \$0.023/GB/month)
- Compute: Expensive (processing, queries)
- Data transfer: Can be costly (egress fees)

Real-Time vs. Batch Data Processing

Batch Processing:

- Process large volumes periodically
- Daily, weekly, monthly updates
- Use cases:
  - Monthly credit score updates
  - Quarterly portfolio rebalancing
  - Annual financial statement analysis
- Tools: Apache Spark, Hadoop, SQL
- Pro: Efficient for large datasets
- Con: Latency (hours to days)

Real-Time (Streaming):

- Process data as it arrives
- Milliseconds to seconds latency
- Use cases:
  - Fraud detection (transaction monitoring)
  - Algorithmic trading (tick data)
  - AML alerts (pattern matching)
- Tools: Apache Kafka, Flink, Kinesis
- Pro: Immediate insights
- Con: Complex infrastructure, costly

Lambda Architecture: Hybrid approach combining batch (accuracy) and streaming (speed) layers.

Data Vendors: Comparison

Bloomberg Terminal:

- Cost: $20-25k/user/year
- Coverage: 99% of financial instruments
- Strengths: Real-time, analytics, news, messaging
- Weaknesses: Expensive, proprietary

Refinitiv (LSEG):

- Cost: $15-30k/year
- Strengths: Historical data, fundamentals
- Eikon platform (Excel integration)

FactSet:

- Cost: $10-15k/year
- Strengths: Quantitative analytics, screening
- Popular among asset managers

Free/Low-Cost Alternatives:

- Yahoo Finance: Free (15-20 min delay)
- Alpha Vantage: API, free tier (500 calls/day)
- Quandl: Free + premium datasets
- FRED (Federal Reserve): Economic data

Alternative Data:

- Thinknum ($10k-50k/year)
- YipitData ($50k+/year)
- Eagle Alpha (marketplace)

Selection Criteria:

- Coverage and accuracy
- Latency requirements
- API availability
- Cost vs. budget

Data Quality Frameworks

Six Dimensions of Quality:

1. **Accuracy:** Values correct?
2. **Completeness:** All required data present?
3. **Consistency:** No contradictions across sources?
4. **Timeliness:** Data current and available when needed?
5. **Validity:** Conforms to format, range, type?
6. **Uniqueness:** No duplicates?

Measurement:

- Accuracy: % records passing validation
- Completeness: % non-null values
- Consistency: % cross-checks passing

Quality Assurance Process:

1. Define quality rules
2. Automated checks (scripts)
3. Exception handling
4. Root cause analysis
5. Continuous monitoring
6. Feedback loop to data sources

Tools:

- Great Expectations (Python library)
- dbt (data build tool)
- Custom SQL checks
- Dashboards (Tableau, Power BI)

Best Practice: Shift-left testing (validate early in pipeline, not at model training).

Summary and Key Takeaways

Data Types:

- Structured (tables) vs. unstructured (text, images)
- Traditional (prices, financials) vs. alternative (satellite, social)
- Real-time (streaming) vs. batch (periodic)

Quality is Critical:

- GIGO principle applies
- 50-80% of effort on data work
- Bias detection and mitigation
- Time series requires special care

Regulations Matter:

- GDPR and privacy laws
- Anonymization challenges
- Right to explanation
- Compliance ¿ model performance

Practical Considerations:

- Cost-benefit of data vendors
- Infrastructure (cloud, storage)
- Feature engineering creativity
- Continuous monitoring

Next Lesson Preview

Lesson 27: Supervised Learning - Regression

Topics to be covered:

- Features, labels, and training data
- Simple and multiple linear regression
- Coefficients interpretation
- R-squared and evaluation metrics
- Overfitting and regularization (Ridge, Lasso)
- Applications: Stock return prediction, pricing models

Preparation:

- Review basic linear algebra (vectors, matrices)
- Recall correlation and covariance concepts
- Think: What financial problems involve predicting continuous values?