AIMLZG565 - Webinar 2

Linear Regression

October 16, 2025

BITS Pilani
Pilani Campus

Disclaimer and Acknowledgement

- The content for these slides has been obtained from books and various other source on the Internet
- I hereby acknowledge all the contributors for their material and inputs.
- I have provided source information wherever necessary
- I have added and modified the content flow to suit the requirements of the course and for ease of class presentation
- Students are requested to refer to the textbook and detailed content of this presentation deck over canvas

Outline

1. Objective/Agenda
2. Why Predictive Maintenance for Cold Storage? (context + challenge)
3. The Problem (Drift consequences, industry-specific risks)
4. Traditional vs Predictive Approaches (table/visual)
5. Workflow Infographic (sensor to action)
6. Data Pipeline: Acquisition & Preprocessing (combine data/sensor and preprocessing topics to reduce repetition)
7. Feature Engineering (include both drift slope and deltas as short visuals)
8. Linear Regression Modeling & Evaluation (combine modeling and evaluation metrics)
9. Case Study: Results & Real-World Impact
10. Key Metrics & ROI (rewrite per above)
11. IoT/Cloud Integration & Real-Time Alerts
12. Case Snippets

Linear Regression for Predictive Maintenance

Predicting Temperature Drift from Sensor Data

Kshitij Dwivedi

Why Predictive Maintenance for Cold Storage?

- **Challenge:**
  - Maintaining optimal temperature is critical to prevent spoilage, weight loss, and quality degradation in shrimp storage.
  - Gradual “temperature drift” (warming over days) often goes unnoticed until thresholds are breached—risking large product losses.
  - Causes include: compressor wear, refrigerant leakage, power instability, insulation failure, or door mismanagement.
- **Consequences:**
  - Undetected drift leads to spoilage, decreased product quality, failed audits, and financial loss.
  - Emergency interventions are costly and disruptive.

The Problem — Temperature Drift in Shrimp Cold Storage

- **Traditional Approach:**
  - Simple alarms for fixed temperature thresholds (e.g., >10°C) react only after a failure or incident.
  - Routine scheduled maintenance can be unnecessary and costly if machines are healthy.
- **Proposed Solution:**
  - Predictive drift analytics: Continuously monitors temperature trend ("drift") to detect early, subtle signs of malfunction.
  - Uses sensor data + linear regression to estimate how quickly temperature is rising—warning issued long before disaster.
- **Why This Solution Works:**
  - Early warning lets staff act *before* storage conditions deteriorate.
  - Enables targeted, data-driven maintenance—saving costs and reducing risk.

Work Infographic Flow

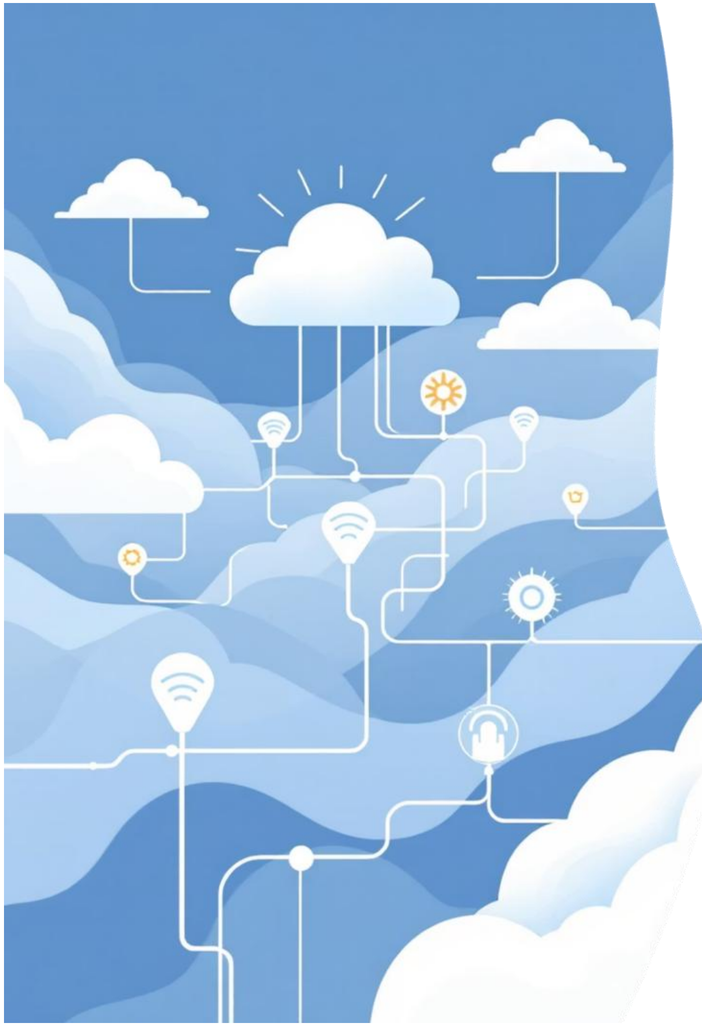
- **Sensor Layer:**
    - PT100 Temperature Sensor
    - Ambient Temperature Sensor
    - (Icons of sensors)

      ↓
- **Connectivity Layer:**
    - PLC / IoT Gateway (Data Collection)

      ↓
- **Data Processing Layer:**
    - Data Storage (Cloud/Edge DB Symbol)
    - Data Preprocessing: Missing data, Outlier removal, Normalizing
    - Feature Engineering: Drift/Slope Calculation

      ↓
- **Analytics Layer:**
    - Linear Regression Model
    - Predict Remaining Useful Life / Identify Temperature Drift
    - Anomaly Detection Alert

      ↓
- **Action Layer:**
    - Real-time Alert (SMS/App)
    - Maintenance Planning Dashboard
    - Scheduling Intervention
    - Key Metrics Dashboard



Data Pipeline: Acquisition & Preprocessing

innovate achieve lead

Data Acquisition

- PT100 temperature sensors installed inside storage; ambient sensors outside.
- Automatic sampling every 10 mins via PLC or IoT gateway.
- Real-time data sent to central server/cloud for storage.

Data Preprocessing

- Missing Data Handling: Interpolated gaps, reindexed timestamps to ensure uninterrupted time series.
- Spike/Outlier Removal: Automatic filtering of unrealistic readings (e.g., 20℃ or large sudden jumps using z-scores/thresholds.
- Synchronization: Align sensor and operational (e.g., compressor state) readings across all records.
- Sanity Checks: Validation of data ranges and sequence before analysis.

Feature Engineering

innovate achieve lead

- **Temperature Drift (Slope):**
  - Calculated rolling slope (linear fit) over past 24 hours.
  - Detects persistent upward trends before absolute limits are crossed.

- **Delta Feature:**
  - Difference between current temperature and value 24h/hours ago.
  - Captures gradual degradation missed by single readings.

Linear Regression Modeling & Evaluation

- **Model Approach:**
  - Linear regression predicts internal temperature as a function of time (and optionally, ambient temp).
  - Slope parameter estimates "drift rate" (°C per hour).
  - Alarms are triggered if the projected threshold crossing is within a warning window.
- **Evaluation Metrics:**
  - Mean Squared Error (MSE): Assesses model fit on test data.
  - Drift Prediction Accuracy: True positives (pre-failure), false alarms.
  - Uptime Impact: Percentage of early warnings issued in advance of actual breaches.

Case Study: Results & Real-World Impact

- **Facility:** Shrimp cold store, multi-week monitoring
- **Key Results:**
  - Early temperature drift alerts enabled preventive maintenance 24–48 hours before failure.
  - 40% downtime reduction (from historical baseline)
  - Lower spoilage incidents: No loss events recorded post-implementation
  - Maintenance efforts focused on real at-risk systems, optimizing resources
- **Business Impact:**
  - Uptime improved, shipments more reliable for export customers
  - Full digital audit trail for compliance and certifications

IoT/Cloud Integration & Real-Time Alerts

- **Data Transmission:** Sensor data automatically uploaded to secure cloud dashboard in real time

- **Analytics & Alerting:**
  - Drift detection algorithm runs continuously on streaming data
  - Alarms pushed to operator smartphones & maintenance dashboards
  - All historical sensor and alert data available for QA, compliance

- **Deployment Overview:**
  - Supports remote cold rooms, low maintenance requirements
  - Integrates easily with most SCADA and ERP platforms

innovate achieve lead

Key Metrics & Impact

- **Productivity Gains**
  - ↑ Process Uptime: 30–40% reduction in cold room downtime
  - ↓ Emergency Maintenance: Up to 25% fewer breakdown interventions
- **Product Quality**
  - Lower Spoilage: 500+ kg/year saved per facility (pilot data)
  - Improved Consistency: Exports meet top-tier compliance more reliably
- **Financial Impact**
  - Cost Savings: ₹10–20 lakh/year from prevented spoilage and avoided crisis repairs
  - ROI: Typical payback in 6–12 months per site
- **Food Safety & Compliance**
  - 99% Compliance Rate: With temperature audit trails and real-time control
- **Sustainability**
  - Less Food Waste: >10% reduction, measurable drop in carbon footprint

How Will It Help the Industry?

- **Risk Mitigation:**
  - Prevents large-scale spoilage events by identifying problems early.
  - Protects inventory and supply chain reliability for exporters/processors.
- **Operational Benefits:**
  - Minimizes unplanned downtime and avoids emergency repairs.
  - Reduces unnecessary maintenance, focusing resources on actual problems.
- **Regulatory/Compliance:**
  - Ensures adherence to hygienic standards and cold chain certifications.
  - Provides digital audit trails with data evidence.

Case snippets

Sensor Data

* Temperature Sensor (PT100)

* Ambient Temperature Sensor (outside the unit,)

* Compressor Power State (on/off)

* Samples every 10 minutes

	timestamp	temp_inside (°C)	ambient_temp (°C)	compressor_on
2	2025-04-01 00:00:00		32.66	0
3	2025-04-01 00:10:00	4.95	33.17	0
4	2025-04-01 00:20:00	5.26	32.18	0
5	2025-04-01 00:30:00	5.62	30.7	0
6	2025-04-01 00:40:00	4.92	32.4	0
7	2025-04-01 00:50:00	4.92	31.35	0
8	2025-04-01 01:00:00	5.65	31.47	0
9	2025-04-01 01:10:00	5.33	32.59	0
10	2025-04-01 01:20:00	4.83	33.24	0
11	2025-04-01 01:30:00	5.24	32.02	0
12	2025-04-01 01:40:00	4.84	32.31	0
13	2025-04-01 01:50:00	4.84	33.7	0

DATA PREPROCESSING

- **Detect and Fill missing timestamps**

```
full_range = pd.date_range(data_missing['timestamp'].min(), data_missing['timestamp'].max(), freq='10min')
data_missing = data_missing.set_index('timestamp').reindex(full_range)
data_missing.index.name = 'timestamp'
```

- **Handle Missing Data**

```
data_missing['temp_inside'] = data_missing['temp_inside'].interpolate()
data_missing['temp_inside'] = data_missing['temp_inside'].bfill()
data_missing['ambient_temp'] = data_missing['ambient_temp'].interpolate()
data_missing['ambient_temp'] = data_missing['ambient_temp'].bfill()
data_missing['compressor_on'] = data_missing['compressor_on'].fillna(0)
```

DATA PREPROCESSING

- **Outlier Filtering - Remove abnormal spikes**

→ *Threshold-based*

```
data_clean = data_missing[(data_missing['temp_inside'] < 20) & (data_missing['temp_inside'] > -5)]
```

→ *z- score based*

```
z = (data_clean['temp_inside'] - data_clean['temp_inside'].mean()) / data_clean['temp_inside'].std()
data_clean = data_clean[abs(z) < 3]
```

FEATURE ENGINEERING

- **Rolling mean for smooth trend**

```
data_clean['temp_rolling'] = data_clean['temp_inside'].rolling(window=12, min_periods=1).mean()
```

- **Rolling slope estimation (drift per day)**

```
data_clean['temp_drift_24h'] = data_clean['temp_inside'].rolling(window=144, min_periods=10).apply(rolling_drift, raw=True)
```

- **Simple drift: difference between current and 24h-ago value**

```
data_clean['temp_drift_delta'] = data_clean['temp_inside'] - data_clean['temp_inside'].shift(144)
```

Linear Regression Modeling

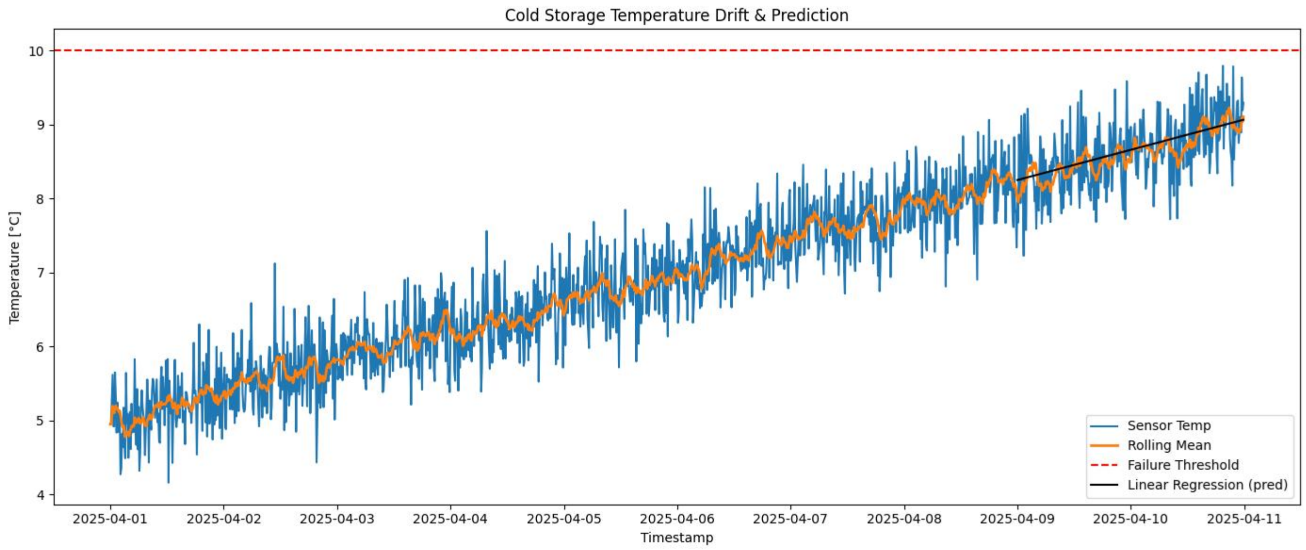
```
X = np.arange(len(data_clean)).reshape(-1, 1)
y = data_clean['temp_inside'].values

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, shuffle=False)
model = LinearRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
slope = model.coef_[0]
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
print(f"Estimated drift rate: {slope:.4f} °C per 10 minutes")
print(f"Test Mean Squared Error: {mse:.3f}")
print(f"Test R^2 Score: {r2:.3f}")
```

Estimated drift rate: 0.0028 °C per 10 minutes
Test Mean Squared Error: 0.164
Test R^2 Score: 0.262

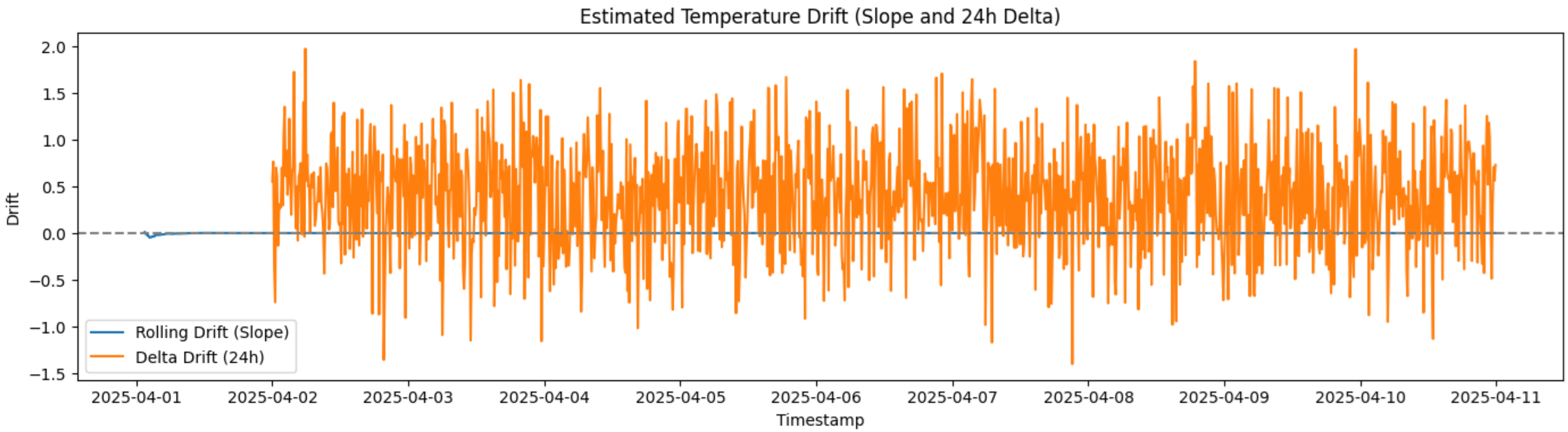
Results: Cold Storage Temperature Drift & Prediction

```
plt.figure(figsize=(14, 6))
plt.plot(data_clean.index, data_clean['temp_inside'], label='Sensor Temp')
plt.plot(data_clean.index, data_clean['temp_rolling'], label='Rolling Mean', linewidth=2)
plt.axhline(10, color='red', linestyle='--', label='Failure Threshold')
plt.plot(data_clean.index[-len(y_test):], y_pred, label='Linear Regression (pred)', color='black')
plt.title('Cold Storage Temperature Drift & Prediction')
plt.xlabel('Timestamp')
plt.ylabel('Temperature [°C]')
plt.legend()
plt.tight_layout()
plt.show()
```



Results: Temperature Drift

```
plt.figure(figsize=(14, 4))
plt.plot(data_clean.index, data_clean['temp_drift_24h'], label='Rolling Drift (Slope)')
plt.plot(data_clean.index, data_clean['temp_drift_delta'], label='Delta Drift (24h)')
plt.axhline(0, color='gray', linestyle='--')
plt.title('Estimated Temperature Drift (Slope and 24h Delta)')
plt.xlabel('Timestamp')
plt.ylabel('Drift')
plt.legend()
plt.tight_layout()
plt.show()
```



innovate achieve lead

Thank you !