# Ensemble Learning for Predicting Key Production Outcomes

SD|Smart Data
SS|Science.Solutions

Isaac Duerr, PhD

21 November 2019

POULTRY® TECH SUMMIT

WATT® GLOBAL MEDIA

# Today's Talk

1. Flexible Technology: Ensemble Machine Learning
2. Use Case: Prediction
3. Use Case: Zero-Cost Experiments
4. Integration With Existing Data Management Systems

# 0. SDSS:
# A Brief Introduction

# What We Do

We provide food producers and related industries with analytical tools that augment industry experience

- Simple: Sampling, analysis, reporting
- Complex: Ensemble Learning, software

# Our Advantage: Flexibility

Flexible Approach

We bring solutions to customers

- No changes to data collection or storage
- No trainings/retrainings
- Little to no IT involvement

# Our Advantage: Flexibility

Flexible Technology

Ensemble Machine Learning

- (The rest of this presentation)

# 1. Flexible Technology: Ensemble Machine Learning

# Machine Learning & Predictive Modeling

## Act, don't react

- Accurate predictions
  - Data
  - Well chosen model
- Enough lead time
  - Timely and accurate data collection and **input**

# Machine Learning & Predictive Modeling

## Act, don't react

- Accurate predictions
  - Data
  - **Well chosen model**
- Enough lead time
  - Timely and accurate data collection and **input**

# Machine Learning & Model Selection

**Optimal model selection is still a question that must be solved for each problem individually**

- There is no single best model
- Model choice affects everything

# Machine Learning & Model Selection

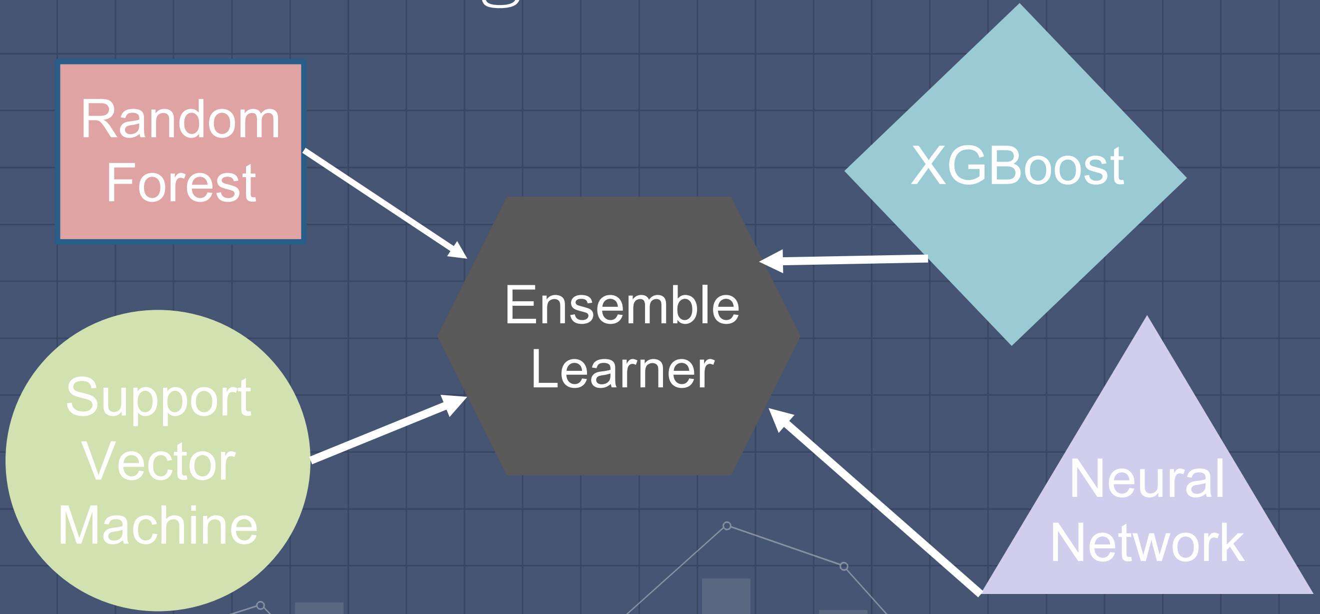
Random Forest

XGBoost

Support Vector Machine

Neural Network

# Machine Learning & Model Selection



# 2. Use Case: Prediction

# What does it mean for Poultry?

- Predict future performance **with enough time to act and correct**
- Understand upcoming trends to capitalize on input cost savings or premium product pricing
- Forecast future production shortages/excesses

# The Data

All Prediction use cases and examples are real

- Live Operations Data
- Location Metadata
- Historical Results

# Use Case 1:
# Predict Key Production Outcomes

- Example 1: Reduce *Salmonella* risk by better planning processing and product mix

- Example 2: Change nutritional packages to improve predicted low-yielding flocks

# Use Case 1:
# Example 1 – *Salmonella*



<u>Example 1: Reduce *Salmonella* risk by better planning processing and product mix</u>

- Generate Risk Score by predicting which flocks are most likely to test positive for *Salmonella* in the **final raw product**, .e.g, tray packs

# Use Case 1: Example 1 - *Salmonella*

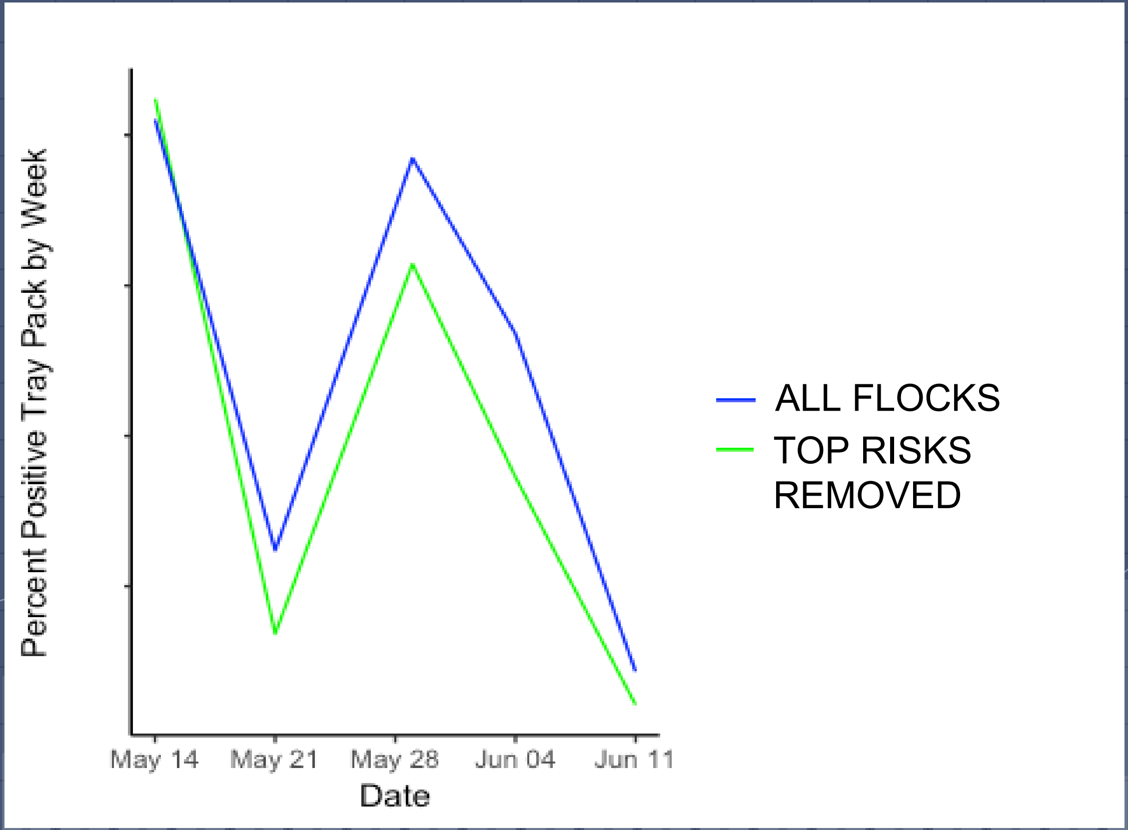


Decisions that drive the timetable:

- Processing Schedule
  - Highest risk flocks processed at end-of-day or end-of-week
- Product Mix
  - Highest risk flocks diverted to cooked, High Pressure Processing, etc.

# Use Case 1:
# Example 1 – *Salmonella*

Results: Estimated weekly reductions of 0-10%



# Use Case 1: Example 2 – Breast Yield



**Example 2: Change nutritional packages to improve predicted low-yielding flocks**

- Generate Yield Score by predicting flock-level breast yield

# Use Case 1:
# Example 2 – Breast Yield

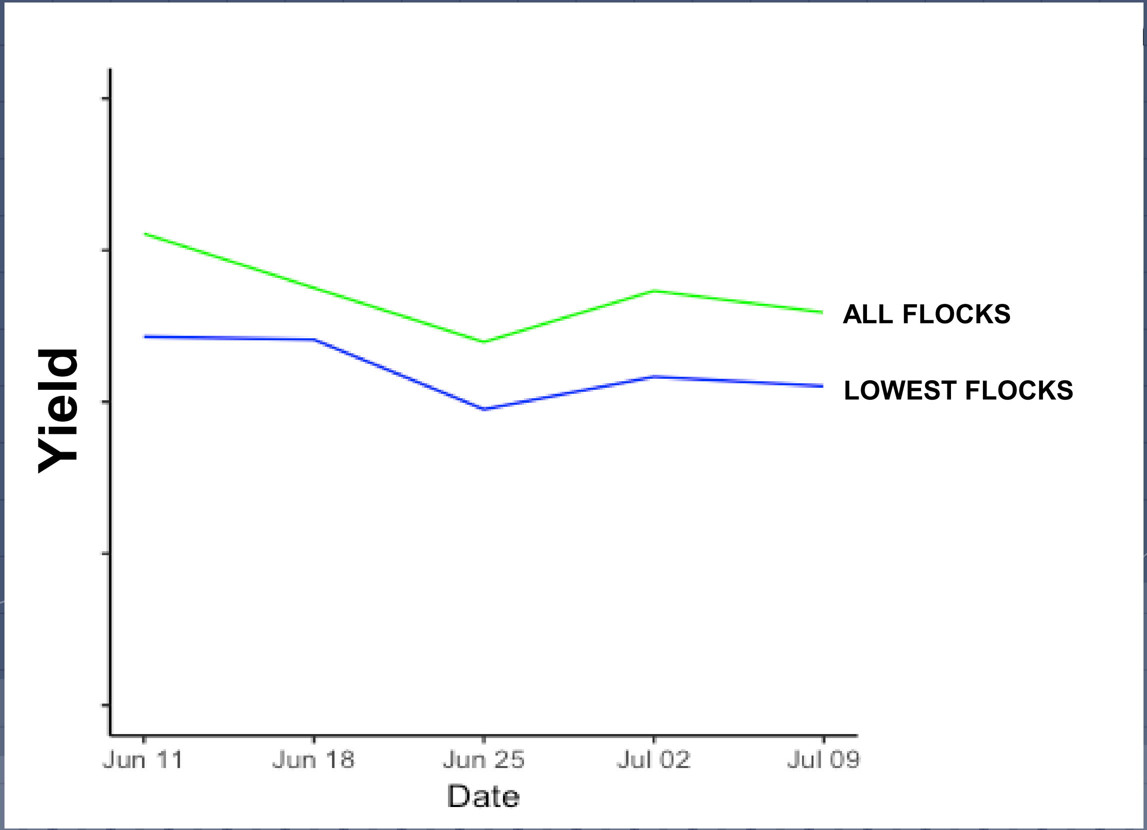
Decisions that drive the timetable:

- Processing Schedule
  - Need enough time pre-processing for a change in nutritional package, H2O, supplements, etc.
- Product Mix
  - When possible, send highest breast yield flocks to premium whole breast products

# Use Case 1: Example 2 – Breast Yield



Results: Predicted lowest-yield flocks are ~10% smaller



# 3. Use Case:
# Zero-Cost Experiments

Zero-Cost Experiments

Ensemble Learning

+

Counterfactual Estimate Generation

# What is a "Counterfactual Estimate"?

An expression of what *did not* happen

Product of

- Epidemiology (necessity)
- Statistics (theory)
- Machine Learning (implementation)

# What does it mean for Poultry?

If we can accurately model a system with EL, we can change inputs to generate an estimate of what *would have happened* to specific outputs of interest

- **Zero-cost experiments**

# Use Case 2: Change Production Parameters

Estimate effects of changing production parameters on key outcomes

- Example: What is the estimated change to overall livability if downtime was increased by 5 days for every flock?

# Use Case 2: Change Production Parameters



## Motivation: Veterinarian vs. Accounting

- The accounting department knows how much an extra day of downtime costs
- The veterinarian believes that the benefit to the birds might pay for itself, if implemented

# Use Case 2: Change Production Parameters

Dataset: Large US chicken company (Top 10)

Data used for analysis was already collected for accounting purposes

- 2000+ Commercial Flocks
- 100+ Farms
- 3 Placement Years

# Use Case 2:
# Change Production Parameters

Experiment:

1. Employ EL to model the system & outcome
2. Add five days to all downtime measures
3. Re-estimate outcome

# Use Case 2:
# Change Production Parameters

## Results:

- Livability + 0.5% overall
  - Over this timeframe
  - For this producer

# 4. Integration With Existing Data Management Systems

# Two Observations

1. Nobody likes changing data management systems, it is not fun for employees
2. Poultry companies don't like changing *anything*, and are especially hesitant to be first adopters

# Solution

Customized cloud-based solutions

- No changes whatsoever to current data collection or management practices
- Accessible to any internet connected device

# Advantages

- Adoption at user's pace
- No square-pegs-in-round-holes problem
  - Each user has what they want in a format they like, with endless possible modular expansion
- Scale up what you want to, when you want to

# Thank you!

# Appendix

# Use Case 2: Estimate Treatment Effects

Estimate treatment effects from three separate treatments

- What would happen to disease status if all animals were given treatment A, B, or C?

# Use Case 2: Estimate Treatment Effects

Motivation: It’s not feasible or practical to use all available treatments, all the time

- Once we have some data about the effects of each treatment, can we better target treatment to individual animals or flocks?

# Use Case 2: Estimate Treatment Effects

Dataset: Undisclosed animal-level experimental dataset

Data was gathered from several locations using no treatment or one or more of the individual treatments

# Use Case 2:
# Estimate Treatment Effects

Experiment:

1. Employ EL to model the system & outcome
2. “Administer” treatment A to all animals
3. Re-estimate outcome
4. Repeat for treatments B and C

# Use Case 2:
# Estimate Treatment Effects



## Results:

| Animal ID | True Health | A | B | C |
|---|---|---|---|---|
| 3665 | Sick | Healthy | Healthy | Sick |
| 1462 | Sick | Sick | Sick | Sick |
| 3470 | Sick | Healthy | Healthy | Sick |
| 2145 | Healthy | Healthy | Healthy | Healthy |
| 2927 | Sick | Sick | Sick | Sick |
| 6441 | Healthy | Healthy | Healthy | Healthy |
| 9028 | Healthy | Healthy | Healthy | Healthy |
| 2492 | Sick | Sick | Sick | Sick |
| 3311 | Healthy | Healthy | Healthy | Healthy |

# Use Case 2:
# Estimate Treatment Effects



## Results:

| Animal ID | True Health | A | B | C |
|---|---|---|---|---|
| 3665 | **Sick** | **Healthy** | **Healthy** | **Sick** |
| 1462 | Sick | Sick | Sick | Sick |
| 3470 | **Sick** | **Healthy** | **Healthy** | **Sick** |
| 2145 | Healthy | Healthy | Healthy | Healthy |
| 2927 | Sick | Sick | Sick | Sick |
| 6441 | Healthy | Healthy | Healthy | Healthy |
| 9028 | Healthy | Healthy | Healthy | Healthy |
| 2492 | Sick | Sick | Sick | Sick |
| 3311 | Healthy | Healthy | Healthy | Healthy |

# Use Case 2: Estimate Treatment Effects

## Results:

- ~20% disease reduction with Treatments A or B
  - Farm-specific
  - Can be expanded to flock-level data

# Use Case 2:
# Monitor Actual vs. Predicted Performance

Identify farms/plants/products that are better or worse **relative to where they should be**

- All producers know their best farms/plants/products in absolute terms
- They don't know which locations are doing better or worse than they *should be*, given their characteristics

# Use Case 2: Monitor Actual vs. Predicted Performance

There is value in identifying which “good” locations should be

- “Great”: Location is **under**performing

Or, conversely, which “good” locations should be

- “Bad”: Location is **over**performing

# Use Case 2: Monitor Actual vs. Predicted Performance

Similarly, there is value in identifying which “bad” locations should be

- “Good” – Location is **under**performing

Or which “bad” locations should be

- “Terrible” – Location is **over**performing

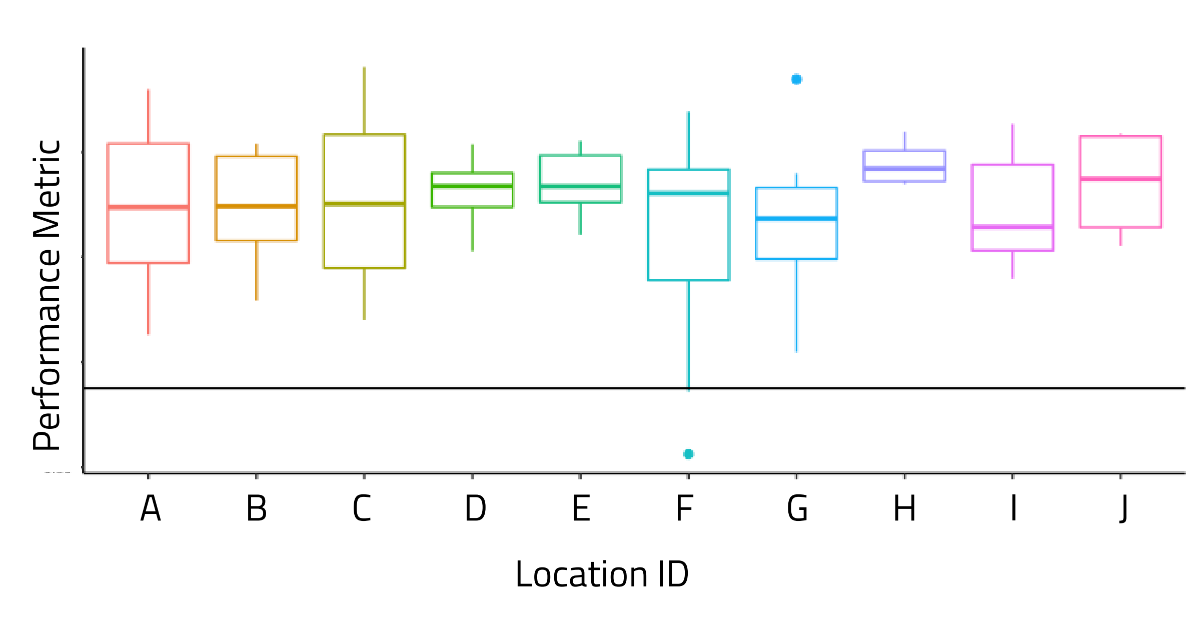
# Use Case 2: Monitor Actual vs. Predicted Performance

Learn what to do (overperforming locations) and what is not working (underperforming)

Track changes over time to see which locations are learning (improving) or forgetting (worsening)

# Use Case 2: Monitor Actual vs. Predicted Performance

## Results: Over-Performing Locations



# Use Case 2: Monitor Actual vs. Predicted Performance

## Results: Under-Performing Locations