Towards Automating Highly Heritable Phenotype Discovery For Plant Breeding

Ruhana Azam¹,², Samuel B. Fernandes³, Mohammed El-Kebir¹, Sanmi Koyejo³, Alexander E. Lipka¹, Andrew D.B. Leakey¹
(¹) University of Illinois at Urbana-Champaign, (²) University of Arkansas, (³) Stanford University

Motivation
• Plant breeders utilize statistical methods (e.g. Genomic Wide Association Study) to predict highly desirable traits in plants.
• Genomic prediction models are trained on biological traits (e.g. nitrogen per leaf area) which are expensive and labor intensive to measure.
• Alternatively, prediction models can be trained using low-cost traits (e.g. hyperspectral data) that are correlated with biological traits of interest. [1]
• Highly-heritable low-cost traits can improve genomic prediction accuracy of high-cost, low-heritability traits [1]

Background
What is heritability?
• Heritability is the portion of population variance explained by genetic factors

\[ h^2 = \frac{\sigma^2_G}{\sigma^2_G + \sigma^2_E + \sigma^2_{\text{error}}} \]

What are synthetic traits?
• Synthetic traits are functional combination of multiple low-cost traits (e.g. \( t_1 / t_2 \), \( t_1 + t_2 / t_3 \))
• Search space grows exponentially with function complexity!

Approach
Black box Search with Bayesian Optimization (BO)
Blind to the heritability function initially and learns the function from querying low-cost traits.

Problem: \( \arg \max_{t_1, t_2, \ldots, t_n} h^2(t_1, \ldots, t_n) \)

How do we discover highly-heritable synthetic traits in large trait spaces?

References

Experimental Setup
Dataset: 836 Sorghum lines, 2 Locations
Baseline: Grid Search (Brute-Force), Random Search
Evaluation: Number of Queries, Time (CPU: Xeon E7-4870)
Synthetic Trait: Wavelength ratios \((t_1 / t_2)\)
Search Space: Smooth in large regions, sparse in others

Preliminary Results
Number of Queries vs Heritability:
Bayesian Optimization methods marginally outperform random search.

Time vs Heritability:
Baseline methods find highly-heritable traits significantly faster than Bayesian Optimization methods.

Conclusion
• When searching wavelength ratio \((t_1 / t_2)\), random search shows best trade-offs over query number and time.
• Bayesian optimization performs well when comparing number of queries.
• Unclear if random search will be exasperated in larger search spaces.

Future Work:
• Expand search to larger function classes (e.g. \( t_1 + t_2 / t_3 \))
• How to deal with search problems where search space and query time are both large?