# Article information:

OutsideIn(X) Modular type inference with local assumptions | Journal of Functional Programming | Cambridge Core  
<https://www.cambridge.org/core/journals/journal-of-functional-programming/article/outsideinx-modular-type-inference-with-local-assumptions/65110D74CF75563F91F9C68010604329>

# Article summary:

1. Advanced type system features, such as GADTs, type classes and type families, are useful for ensuring data invariants and program correctness but pose a difficult problem for type inference when used as local type assumptions.

2. Local let-binding generalisation is often not used by Haskell programmers and is difficult to implement and specify.

3. OutsideIn(X) is a novel constraint-based type inference approach for local type assumptions that relies on a constraint solver for X and only accepts programs with principal types.

# Article rating:

May be slightly imbalanced: The article presents the information in a generally reliable way, but there are minor points of consideration that could be explored further or claims that are not fully backed by appropriate evidence. Some perspectives may also be omitted, and you are encouraged to use the research topics section to explore the topic further.

# Article analysis:

The article “OutsideIn(X) Modular Type Inference with Local Assumptions” provides an overview of the challenges posed by local type assumptions in advanced type systems, as well as a novel approach to solving them. The article is written in an objective manner and presents both sides of the argument fairly, making it trustworthy and reliable overall.

The authors provide evidence to support their claims, such as empirical results showing that local let generalisation is rarely used by Haskell programmers, as well as references to other research papers on related topics. They also present a detailed description of their proposed solution – OutsideIn(X) – which includes a particular constraint solver for X = type classes + GADTs + type families. This makes the article highly informative and reliable in terms of its content.

However, there are some potential biases in the article that should be noted. For example, the authors focus mainly on the benefits of their proposed solution without exploring any potential drawbacks or counterarguments. Additionally, they do not discuss any possible risks associated with using their approach or how it might affect existing code bases or programming practices. These points should be considered when evaluating the trustworthiness of this article.

# Topics for further research:

* Risks associated with type inference
* Impact of type inference on existing code bases
* Drawbacks of local let generalisation
* Type classes + GADTs + type families
* Potential issues with OutsideIn(X) approach
* Alternatives to OutsideIn(X) type inference

# Report location:

<https://www.fullpicture.app/item/d2878ff3c55f23bc7b318172ecc0f29b>