Bringing AI to DuckDB with Lance columnar format for multi-modal AI
• LanceDB co-founder
• Pandas co-author
• VP Eng @ TubiTV (MLOps / Experimentation)
• Twitter/Github: @changhiskhan
# Lance columnar format

High performance unified storage for AI

<table>
<thead>
<tr>
<th>Unified storage</th>
<th>Plug-and-play</th>
<th>Reduce cost</th>
<th>Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Store embeddings, text, images, pdfs, videos, audio, point clouds, alongside tabular data.</td>
<td>Convert data with 2 lines of code. Compatible with pandas, polars, duckdb, spark, jupyter, and more</td>
<td>Reduce storage cost by 80%. Replaces parquet, tfrecords, etc. Don't need multiple copies in different formats. Zero-copy schema evolution.</td>
<td>Reduce training time by up to 3x with faster filtering, shuffling, and data loading. Up to 2000x faster than parquet for AI</td>
</tr>
</tbody>
</table>
Bring AI into DuckDB ecosystem

```sql
SELECT filename, class, predict('resnet', image) as pred
FROM oxford_pet
WHERE split='train' AND class='samoyed'
USING SAMPLE 1000;
```
Bring AI into DuckDB ecosystem

- Data: Lance extension
- Model: Pytorch / TF extensions
- Scanners: e.g., ffmpeg/opencv scanner for videos
- UDFs: e.g., crop, rotate, sample images, etc
- CUDA integration
Types! Types! Types!

- Composable data systems is not a thing unless types are more interoperable.

- Arrow types and DuckDB types are maybe 80% interoperable. Unfortunately AI falls in the missing 20%.
  - Nested types - annotations, bounding boxes, labels, etc
  - Extension types - image, embedding, video, point cloud, etc
  - ML specific types - e.g., bf16
DuckDB-Arrow pushdowns

- pyarrow compute kernel is not a standard interface
- Lance format uses datafusion so can pushdown sql str
- Maybe long-term use Substrait as the standard interface across Arrow-compatible formats?
- Lance format has a disk-based vector index, but duckdb / pandas / polars can't access it directly because there's no pushdown available yet.
  - Table function in an extension is viable.
  - Better would be to allow extension to modify the query plan (order by / limit)
Conclusion

- Find us on github: github.com/lancedb/lance
- New data format for AI is needed
- Need a number of improvements to bridge AI into DuckDB ecosystem
- Looking for collaborators!