Overview
Neural networks have been widely responsible for recent advances in machine learning, powering technologies such as digital assistants and AR photography. LPLANN (Low-Precision Linear Algebra for Neural Networks) is a cross-platform library written in C++ used for implementing neural networks. The software allows users to set specific levels of precision for calculations. Low-precision calculations use advanced parallelization techniques (SIMD, SWAR) to run neural networks at faster rates than full-precision calculations. This library is lightweight enough to run on embedded systems, but also includes optimizations that would provide drastic speedups on a workstation. LPLANN also serves as a testbed for novel low-precision neural network architectures.

Problem
Part of the reason deep neural networks have been so successful is because of advances in GPU technology which allows for faster training and performance. However, GPUs are expensive and use a huge amount of power. By using SIMD within a register (SWAR), LPLANN can quickly execute low-precision neural networks on a CPU. This will allow neural networks to be executed on a embedded systems or other constrained devices.

Results
LPLANN is a working platform for implementing neural networks on systems without a GPU. It offers drastic speedups by using multithreading and SIMD within a register. The table below shows frames per second (FPS) running on Raspberry Pi B. The binary network runs at more than double the speed of the full precision network. Depending on network architectures, even faster speedups are possible.

<table>
<thead>
<tr>
<th>Precision</th>
<th>FPS</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>32-bit</td>
<td>0.378</td>
<td>1.29 billion</td>
</tr>
<tr>
<td>Binary</td>
<td>0.935</td>
<td>214 million</td>
</tr>
</tbody>
</table>

Low-precision types
LPLANN can perform operations at any of the precision levels shown here, and can convert freely between precision levels.

Convolutional neural networks
Modeled after the human visual system, convolutional or “deep” neural networks are used to perform tasks that are intuitive for humans, but very complex to program, such as understanding speech or recognizing objects in photos.

Filters are trained by the network to identify certain features. More complex features are learned by later network layers, then those features are used to identify an input.

Working with binary values
The most drastic optimization offered by LPLANN is how it operates on multiple binary weights simultaneously. By placing values in a single register, the XOR operation acts as binary multiplication, performing many operations with a single instruction.

Acknowledgements
Special thanks to Don Cripps, Jake Gunther, and Jolynne Berrett