

Assignment Submission Form

1. Group Code/Name

eldad_ron_bar_yacobi

2. Team Member 1

ID (Required): 207021916

Name (Optional): Eldad Ron

3. Team Member 2

ID (Required): 315471367

Name (Optional): Bar Yacobi

4. GitHub Repository Link

Repository URL:

<https://github.com/er1009/LLMs-And-Multi-Agent-Orchestration-Course/tree/main/ex2>

*Note: Please ensure the repository is set to **Public** for grading purposes.*

5. Self-Recommended Grade

Recommended Grade: 100

Justification:

We recommend a grade of 100 based on the following:

Technical Implementation (30 points):

- Complete LSTM neural network implementation for conditional signal extraction
- Proper state management for L=1 sequence length (critical requirement)
- Full ML pipeline: data generation → training → evaluation → visualization
- Synthetic signal generation with configurable parameters and random seeds
- Conditional regression using one-hot encoded frequency selector
- Support for multiple devices (CPU, CUDA, MPS) with auto-detection
- Gradient accumulation for efficient training
- Learning rate scheduling with ReduceLROnPlateau

Documentation (25 points):

- Comprehensive README.md with detailed installation and usage instructions
- Complete PRD (Product Requirements Document) with all required sections
- Architecture Document (ARCHITECTURE.md) with C4 model, ADRs, and technical details
- Professional development documentation (PRD_PROMPT.md, AI_PROMPTS.md, DEVELOPMENT.md)
- Clear project structure documentation
- Comprehensive troubleshooting section
- All requirements clearly addressed and documented

Testing (20 points):

- Unit tests for dataset generation (test_dataset.py)
- Unit tests for LSTM model (test_model.py)
- Integration tests for end-to-end pipeline (test_integration.py)
- All tests include expected results in docstrings
- Tests verify seed reproducibility, shape validation, and model learning
- Tests cover edge cases and generalization checking
- Total of 7+ tests covering all major functionality
- All tests pass successfully

Code Quality (15 points):

- Single responsibility principle followed throughout
- Clean, modular architecture (data, model, training, evaluation, visualization)
- Comprehensive docstrings for all modules, classes, and public functions
- Comments explain 'why' not just 'what'
- DRY principles followed, no unnecessary complexity
- Consistent naming conventions and formatting
- Proper error handling and graceful device fallback
- Well-structured project organization

Results & Visualization (10 points):

- All required visualizations generated (4 plots as specified)
- Single frequency comparison plot (Graph 1)

- All 4 frequencies extraction plot (Graph 2)
- Training history visualization
- MSE comparison plot (train vs test)
- High-quality plots (300 DPI) with proper labels and legends
- Model achieves good generalization ($MSE_{test} \approx MSE_{train}$)
- Results demonstrate successful frequency extraction

Requirements Compliance:

- Uses virtual environment (conda/venv) and requirements.txt
- Complete LSTM implementation with proper state management
- Conditional regression using one-hot frequency selector
- Dataset generation with different seeds for train/test
- MSE calculation on both training and test sets
- Generalization checking implemented
- All assignment requirements met
- Professional documentation structure
- Complete PRD and Architecture documents included
- All unit tests with expected results documented

Additional Strengths:

- Comprehensive architecture documentation with C4 model and ADRs
- Proper .gitignore file for version control
- Support for sequence length $L > 1$ (extensibility)
- Optional dataset caching for efficiency
- Cross-platform compatibility (macOS, Linux, Windows)
- Professional code quality meeting M.Sc. standards
- Complete implementation of all required features
- Clean, maintainable codebase with excellent separation of concerns
- All deliverables completed to specification

We believe this submission demonstrates complete understanding of LSTM architecture, proper state management, conditional regression, and professional software engineering practices. The project fully meets all assignment criteria and exceeds expectations in documentation quality and code organization.

6. Special Notes

None

7. Attached Special Documents

None

8. Comments Section

(This page is left blank for instructor comments)