# 1 Sobre os Trabalhos

## 1.1 Objetivo

O objetivo do trabalho é realizar um projeto sobre a análise e implementação de um tema abordado nas aulas com uma parte de programação e de experiência numérica.

- Serão mantidos os grupos já formados no âmbito da UC Projeto Integrado bem como o representante do grupo;
- Terão de entregar um relatório com 10-15 páginas e a apresentação
- A classificação deste trabalho será: 50% do relatório, 50% da apresentação oral.
- Cada grupo deve escolher uma proposta na lista apresentada mais à frente.

## 1.2 Procedimentos

1. **Dia 23/04, a partir de 15h00**, cada representante do grupo envia-me um email (mfc@math.uminho.pt) com a ordem de preferência dos projetos a realizar. Vamos atribuir os projetos consoante a hora de chegada dos emails (excluindo os emails que chegarem mais cedo).
2. Após terem conhecimento do projeto que vão efetivamente realizar, o representante do grupo terá que contactar o professor/orientador para a a organização de sessões de trabalho.
3. O representante deve enviar por email, os pdf do relatório e da apresentação ao seu orientador **até 30 maio às 18:00**.
4. As apresentações serão no **dia 3 de junho, das 9:00 às 13:00** na sala das aulas (20 minutos para apresentação e 30 minutos para discussão, por grupo).

# 2 Temas

Os temas, detalhados no resto do documento, são o seguintes.

- T1 Classificador logístico multiclasse: OvA vs ECOC
- T2 Classificador logístico multiclasse: OvO vs ECOC
- T3 Soft margins SVM: SG vs ADAM
- T4 Soft margins SVM: SG vs RMSProp

## 2.1 Classificador logístico multiclasse: OvA vs ECOC [Orientador GJM]

**Descrição.** Propõe-se neste trabalho comparar (tempo computacional e eficiência da aprendizagem) as técnicas *one-vs-all* (OvA) e *Error-Correcting Output Codes* (ECOC) do classificador logístico multi-classe nas versões primal e dual com *kernel* polinomial. Deve-se testar diferentes regras de aprendizagem. Deve-se também considerar a versão *mini-batch* com diferentes tamanhos (desde um até ao tamanho da base de dados) e uma versão *"mini-batch"* com tamanho um

mas em que a base de dados é percorrida sequencialmente. Inicialmente deve-se considerar bases de dados sintéticas com três classes com características diferentes mas apenas com dois atributos ($I = 2$) por forma a se poder fazer a visualização dos resultados. Deve-se depois considerar uma base de dados real com pelo menos quatro classes. A avaliação dos resultados deve incluir diferentes indicadores, um dos quais a matriz de confusão. O grupo pode escolher a linguagem de programação que vai usar.

**Trabalhos a realizar.**

1. Descrição e implementação do método primal com as versões OvA e ECOC.
2. Validação e avaliação com as bases de dados.
3. Descrição e implementação do método dual com as versões OvA e ECOC.
4. Validação com as bases de dados.
5. Descrição e implementação do método dual com *kernel* polinomial com as versões OvA e ECOC.
6. Validação e avaliação com as base de dadoss.

O documento de referência é `http://ciml.info/dl/v0_99/ciml-v0_99-all.pdf`, nomeadamente as secções 6.2 e 11, e `http://www.ccs.neu.edu/home/vip/teach/MLcourse/4_boosting/lecture_notes/ecoc/ecoc.pdf`.

## 2.2 Classificador logístico multiclasse: OvO vs ECOC [Orientador GJM]

**Descrição.** Propõe-se neste trabalho comparar (tempo computacional e eficiência de aprendizagem) as técnicas *one-vs-one* (OvO) e *Error-Correcting Output Codes* (ECOC) do classificador logístico multi-classe nas versões primal e dual com *kernel* polinomial. Deve-se testar diferentes regras de aprendizagem. Inicialmente deve-se considerar bases de dados sintéticas com três classes com características diferentes mas apenas com dois atributos ($I = 2$) por forma a se poder fazer a visualização dos resultados. Deve-se depois considerar uma base de dados real com pelo menos quatro classes. A avaliação dos resultados deve incluir diferentes indicadores, um dos quais a matriz de confusão. O grupo pode escolher a linguagem de programação que vai usar.

**Trabalhos a realizar.**

1. Descrição e implementação do método primal com as versões OvO e ECOC.
2. Validação e avaliação com as bases de dados.
3. Descrição e implementação do método dual com as versões OvO e ECOC.
4. Validação com as bases de dados.
5. Descrição e implementação do método dual com *kernel* com as versões OvO e ECOC.
6. Validação e avaliação com as bases de dados.

Os documentos de referências são `http://ciml.info/dl/v0_99/ciml-v0_99-all.pdf`, nomeadamente as secções 6.2 e 11, e `http://www.ccs.neu.edu/home/vip/teach/MLcourse/4_boosting/lecture_notes/ecoc/ecoc.pdf`.

### 2.3 Soft margins SVM: SG vs ADAM [Orientador FC]

**Descrição.** Propõe-se neste trabalho o estudo e implementação Soft Margins SVM (C-SVM) na versão dual, sem e com *kernel*. Para as bases de dados a considerar, deverá considerar dois valores possíveis para o parâmetro C e algumas funções para o *kernel*. Propõe-se neste trabalho comparar os métodos Stochastic Gradient (SG) e Adam Stochastic (tempo computacional e eficiência de aprendizagem) no treino do C-SVM dual.
**Trabalhos a realizar.**

1. Descrição do método C-SVM dual.
2. Reformulação do problema C-SVM dual para treino com o SG.
3. Descrição e implementação do algoritmo de treino do C-SVM dual com SG (sem e com função kernel).
4. Descrição e implementação do algoritmo de treino do C-SVM dual com Adam Stochastic (sem e com função kernel).
5. Validação e avaliação com as bases de dados, do classificador C-SVM dual sem *kernel* obtido com SG *versus* Adam Stochastic.
6. Validação e avaliação com as bases de dados, do classificador C-SVM dual com *kernel* obtido com SG *versus* Adam Stochastic.

Os documentos de referências são: `https://dataminingbook.info/book_html/chap21/book.html`, nomeadamente as seções 21.3.1, 21.4, e 21.5

### 2.4 Soft margins SVM: SG vs RMSProp [Orientador FC]

**Descrição.** Propõe-se neste trabalho o estudo e implementação Soft Margins SVM (C-SVM) na versão dual, sem e com *kernel*. Para as bases de dados a considerar, deverá considerar dois valores possíveis para o parâmetro C e algumas funções para o *kernel*. Propõe-se neste trabalho comparar os métodos Stochastic Gradient (SG) e RMSProp Stochastic (tempo computacional e eficiência de aprendizagem) no treino do C-SVM dual.
**Trabalhos a realizar.**

1. Descrição do método C-SVM dual.
2. Reformulação do problema C-SVM dual para treino com o SG.
3. Descrição e implementação do algoritmo de treino do C-SVM dual com SG (sem e com função kernel).
4. Descrição e implementação do algoritmo de treino do C-SVM dual com RMSProp Stochastic (sem e com função kernel).
5. Validação e avaliação com as bases de dados, do classificador C-SVM dual sem *kernel* obtido com SG *versus* RMSProp Stochastic.
6. Validação e avaliação com as bases de dados, do classificador C-SVM dual com *kernel* obtido com SG *versus* RMSProp Stochastic.

Os documentos de referências são: `https://dataminingbook.info/book_html/chap21/book.html`, nomeadamente as seções 21.3.1, 21.4, e 21.5