

# ZMiTAD 2015/2016 letni

Łukasz Augustyniak, Roman Bartusiak

Luty 2016

## 1 Warunki zaliczenia

### 1.1 Przebieg zajęć

Istnieje możliwość zaliczenia zajęć wykorzystując dwie ścieżki:

1. Listy zadań + mini projekt
2. Projekt

### 1.2 Listy zadań + mini projekt

Na kursie realizowane będą zadania związane z materiałami podanymi na wykładzie w programach Weka oraz Matlab (Octave).

#### 1.2.1 Warunki zaliczenia

1. Nie więcej niż jedna nieusprawiedliwiona nieobecność w semestrze. Usprawiedliwienie należy przedstawić na pierwszych zajęciach bezpośrednio po nieobecności.
2. Zgromadzenie ponad 50% punktów za wykonane w ramach kursu zadania.
3. Ukończenie mini-projektu

#### 1.2.2 Ocenianie studentów

Na ocenę wystawioną przez prowadzącego składają się następujące elementy:

1. Wykonywanie ćwiczeń zadanych przez prowadzącego.
2. Wiedza teoretyczna (**sprawdzana** podczas prezentacji list na zajęciach)
3. Za każdą listę student może otrzymać maksymalnie 5 pkt (w sumie 11\*5pkt + 10pkt za projekt)
4. Terminowość prezentowania rozwiązań list zadań
  - Wysłanie rozwiązania na adres mailowy prowadzącego (najpóźniej do początku zajęć, na których prezentowana jest dana lista)
  - Zaprezentowanie rozwiązania na zajęciach.
  - Termin wysłania jest określony dla każdej listy i jest to termin nieprzekraczalny.
  - W przypadku nieobecności na zajęciach student do rozpoczęcia zajęć, na których oddawana jest dana lista zadań powinien wysłać jej rozwiązanie mailem do prowadzącego oraz zaprezentować rozwiązanie na pierwszych zajęciach po nieobecności. W przeciwnym przypadku lista nie będzie sprawdzana
  - Za listę wysłaną po terminie lub niezaprezentowaną student otrzymuje 0 punktów.

Istnieje możliwość uzyskania dodatkowych punktów (10pkt) poprzez:

- Prezentację na uzgodniony w prowadzącym **INTERESUJĄCY** i przydatny temat. Prezentacja musi zawierać opis oraz praktyczne zastosowanie wybranego zagadnienia.
- Ukończenie zadania konkursowego Kaggle.

Punkty	Ocena
[0%, 50%]	2
[60%, 60%]	3
(60%, 70%]	3.5
(70%, 80%]	4
(80%, 90%]	4.5
(90%, 100%]	5

Tabela 1: Skala ocen przy zaliczeniu zajęć przez listy zadań

## 1.3 Projekt

Realizacja projektu na wybrany temat w grupach *maksymalnie* 3 osobowych. Projekt powinien kończyć się swego rodzaju produktem.

### 1.3.1 Warunki zaliczenia

1. Przygotowanie harmonogramu prac.
2. Regularne prezentowanie postępów prac zgodne z harmonogramem.
3. Czynny udział w pracy każdego członka zespołu.
4. Zakończenie każdego etapu (prezentacja pracy na zajęciach) poprzedzone powinno być wstępem teoretycznym
5. Oceniane jest zakończenie każdego etapu projektu
6. Korzystanie z repozytorium kodu (BitBucket, Github - Student Pack<sup>1</sup>).

### 1.3.2 Ramowy plan projektu

- Zdefiniowanie problemu badawczego (hipoteza badawcza)
- Przygotowanie harmonogramu
- Przygotowanie środowiska
- Pozyskanie danych
  - Wykorzystanie API lub innych metod (scraping itd.) jest dodatkowym plusem przy ocenie ostatecznej.
  - Wykorzystanie gotowego zbioru danych wymusza poszerzenie zakresu projektu (np. duży zbiór danych, więcej analiz)
- Czyszczenie i transformacja
- Wstępne analizy statystyczne (rozkłady itd.)
- Usuwanie wartości odstających (ang. outliers)
- Budowanie modelu uczenia maszynowego (kilku)
- Ewaluacja modeli
- Wizualizacja wyników
- Prezentacja

---

<sup>1</sup><https://education.github.com/pack>

### 1.3.3 Ocenianie studentów

Na ocenę składa się wiele czynników, między innymi:

- Terminowość
- Systematyczność
- Zgodność z harmonogramem
- Wiedza teoretyczna
- Podział pracy
- **SAMODZIELNOŚĆ** wykonania zadań

### 1.3.4 Proponowane tematy:

1. Sieć projektów na podstawie danych z GitHub'a - wykrywanie wartości odstających (ang. outlier) <http://ghtorrent.org>
2. Analizy strumienia zdarzeń w czasie rzeczywistym - Github Events API
3. Analiza danych tekstowy w czasie rzeczywistym - Twitter API
4. Analizy danych o społeczności OpenSource ze względu na lokalizację - Github Events API, GIS
5. Analiza danych Wrocławskiego Budżetu Obywatelskiego
6. Analiza danych o publikacjach i cytowaniach <https://www.kaggle.com/benhamner/d/benhamner/nips-2015-papers/exploring-the-nips-2015-papers/notebook>
7. Analiza wydźwięku lub emocji występujących artykułach prasowych - Event Registry <http://eventregistry.org/searchEvents>
8. Predykcja rynków finansowych, ang. Forecast Financial Markets - (np. na podstawie Event Registry, historyczne kursy walut)
9. Projekty na podstawie danych wrocławskich : <http://www.wroclaw.pl/open-data/>
10. Projekty na podstawie danych z platformy Kaggle: <https://www.kaggle.com/datasets>
11. Projekty na podstawie danych EU: <http://www.europeandataportal.eu/>
12. Rekomendacja repozytoriów dla deweloperów (GHTorrent, GitHub Archive)
13. Systemy rekomendacyjne , np. wykorzystanie danych z <https://www.kaggle.com/c/event-recommendation-engine-challenge/data>
14. Analiza języka naturalnego, np. wykrywanie tematu wypowiedzi, atrybutów opisujących produkty występujące w tekście/opiniach itp.
15. *Propozycje studentów*

\* Tematy projektów mogą ulec zmianie po uzgodnieniu z prowadzącym

## 2 Literatura

1. **Advances in Social Media Analysis**, edited by Mohamed Medhat Gaber, Mihaela Cocea, Nirmalie Wiratunga, Ayse Goker
2. **Mining the Social Web, 2nd Edition, Data Mining Facebook, Twitter, LinkedIn, Google+, GitHub, and More**, Matthew A. Russell, O'Reilly Media, October 2013

### 3 Przydatne źródła

1. <https://www.coursera.org> (np. <https://www.coursera.org/learn/machine-learning>)
2. <https://www.import.io>
3. <http://scrapinghub.com/opensource>
4. <https://www.kaggle.com>
5. <https://realpython.com/blog/python/twitter-sentiment-python-docker-elasticsearch-kibana> pomysły na wizualizacje real-time